kilobaud

September 1981
USA $2

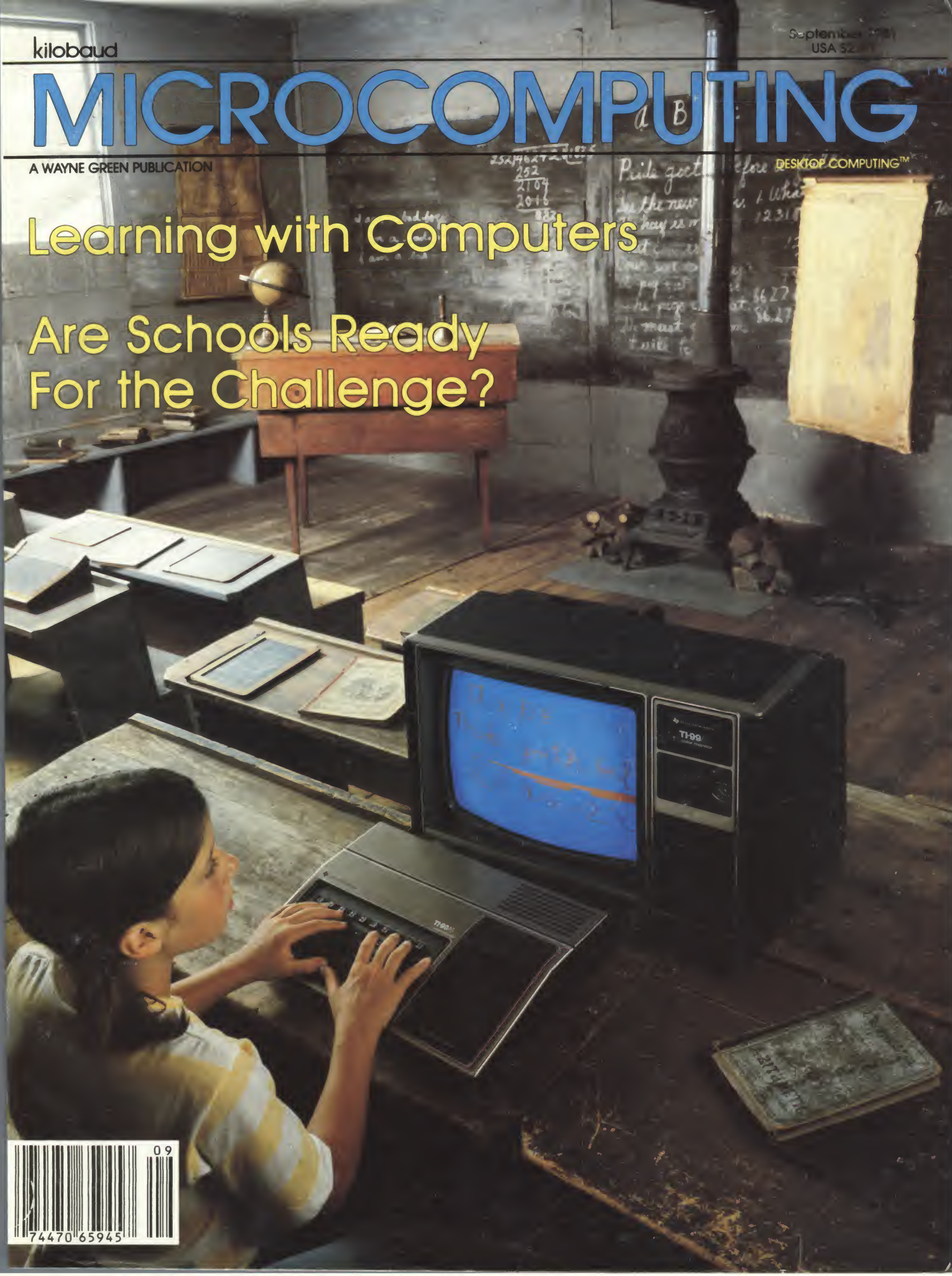
MICROCOMPUTING™

A WAYNE GREEN PUBLICATION

DESKTOP COMPUTING™

Learning with Computers

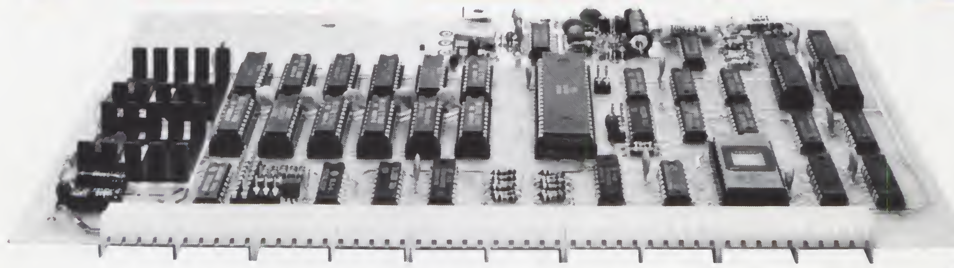
Are Schools Ready For the Challenge?



74470 65945 09

Now! Color for Your . . .

SYSTEM-50

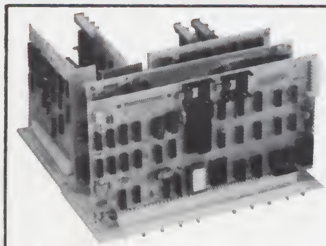


Introducing COLO*RAMA*-50™
Percom's SS-50 Bus Color VDG

Introductory Price $219.95

Featuring . . .

- **Eleven display formats** including 8-color semigraphics, 4-color graphics, 2-color high density graphics and 2-color alphanumerics.
  Moreover, two- and four-color displays may be switched between primary and complementary color sets under software control or from the keyboard.
  Full graphic resolutions range from 64 x 64 picture elements to 256 x 192 picture elements.
- **Instant display control:** The COLO*RAMA*-50™ is memory mapped: your MPU has direct, instant access to display RAM and display control registers.
- **Low-cost Modulator Option for Color TV Interface:** The COLO*RAMA*-50™ provides for installation of an inexpensive RF modulator such as Radio Shack PN 277-122 for operation using a color TV.



SS-50 Bus Department Store

Nobody supports the SS-50 bus like Percom:

- ✓ SS-50 Bus/Single-Board Computers with I/O ports & memory
- ✓ Static and Dynamic RAM cards — memory expansion kits
- ✓ LFD-400/800 1-, 2- and 3-drive mini-disk systems
- ✓ Color and monochrome memory-mapped display controllers
- ✓ Extendable 7-slot SS-50 bus motherboards
- ✓ Versatile prototyping boards: SS-50 and SS-30 bus
- ✓ Field-proven software: monitors, operating systems, drivers, editors, assemblers, debuggers and HLLs.

- **Mix in Sound:** With the optional modulator installed, you can complement your colorful displays with software-controlled audio.
- **Extended Addressing:** The COLO*RAMA*-50™ is compatible with the SS-50A bus *and* the extended-address SS-50C bus. Map the board into any of the sixteen 64-Kbyte banks of the 1-Mbyte SS-50C address space. The COLO*RAMA*-50™ card "defaults" to the first (lowest) bank for the SS-50A bus.
- **Cassettee I/O Option:** Add a few inexpensive components to the on-card circuitry provided and use an audio cassette for program/data storage.
- **Provision for On-Card Firmware:** Put your display operating system, cassette control program, etc. right on the COLO*RAMA*-50™ card in a 2516 (5-volt 2716) EPROM. Resides in the top 2-Kbyte of the card memory space.
- **Operating Software:** Included in the comprehensive users manual is a listing of a display operating system and cassette controller that may be implemented as a callable subroutine function from BASIC or existing operating systems. The programs are optionally available in a plug-in ROM for just $69.95.

System Requirements

The COLO*RAMA*-50™ is pin- and outline-compatible with the Percom System-50™ bus, the SS-50A (SS-50) bus and the SS-50C bus. The composite video-sync signal output will directly drive a color (or BW) video monitor. The output may be modulated for operation with a standard (NTSC) TV set. A modulator is not included. The COLO*RAMA*-50™ card occupies 8-Kbytes of memory in the upper half of a 64-Kbyte memory space. Included on-card is 1-Kbyte of display RAM which will accommodate alphanumeric displays, semigraphic displays and two low-density full-graphic displays. For the higher density graphic displays, additional display RAM is required. The optional RAM ICs may be installed on the card.

For quality Percom SS-50 bus products, see your nearby authorized Percom dealer. To order direct, call **toll-free, 1-800-527-1592.** Prices and specifications subject to change without notice. Prices do not include shipping and handling.

PERCOM DATA COMPANY, INC.
211 N. KIRBY GARLAND, TEXAS 75042
(214) 272-3421

PERCOM

™ trademark of Percom Data Company, Inc.

MICROCOMPUTING

PUBLISHER/EDITOR
Wayne Green
EXECUTIVE VICE PRESIDENT
Sherry Smythe
EDITORIAL MANAGER
Jeff DeTray
PUBLICATIONS MANAGER
Edward Ferman
MANAGING EDITOR
Dennis Brisson
ASSISTANT MANAGING EDITOR
Susan Gross
COPY EDITOR
Eric Maloney
TECHNICAL EDITORS
Harold Nelson
G. Michael Vose
EDITORIAL ASSISTANTS
Lise Markus, Linda Stephenson, Betty Thayer
ADMINISTRATIVE ASSISTANTS
Pat Graham, Nancy Noyd
ASSOCIATE EDITORS
Robert Baker, Ken Barbier, Frank Derfler, Jr., Rod Hallen, Peter Stark, Sherm Wantz

PRODUCTION MANAGER/PUBLICATIONS
Nancy Salmon
ASSISTANT PRODUCTION MANAGER
Michael Murphy
ART DIRECTOR
Diana Shonk
ADVERTISING GRAPHICS
Steve Baldwin, Dennis Christensen, Robert Drew, Bruce Hedin, Jane Preston
PRODUCTION DEPARTMENT
Joan Ahern, Fiona Davies, Linda Drew, Bob Dukette, Sandra Dukette, Gary Graham, Kenneth Jackson, Ross Kenyon, Pat Mackowsky, Theresa Ostebo, Sharon Phinney, Dianne Ritson, Deborah Stone, Susan Symonds, Judi Wimberly, Donna Wohlfarth
PHOTOGRAPHY
Terrie Anderson, Paul Babich, William Heydolph, Bill Suttenfield, Thomas Villeneuve
TYPESETTING
Sara Bedell, Michele DesRochers, Stephen Jewett, Luann Keddy, Mary Kinzel, Kelly Smith, Karen Stewart

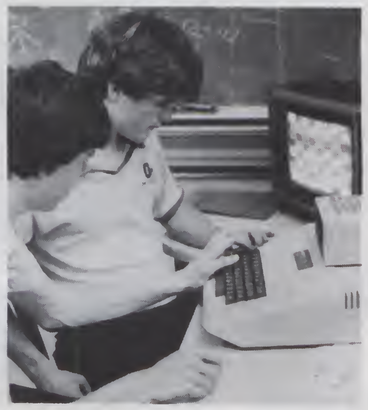
CORPORATE CONTROLLER
Charles Garniss, Jr.
EXECUTIVE ASSISTANT
Leatrice O'Neil
ACCOUNTING MANAGER
Knud Keller
DIRECTOR OF MARKETING
603-924-7296
Debra Boudrieau
DIRECTOR OF CORPORATE ADVERTISING
Elaine Cheever
CIRCULATION
Doris Day, Pauline Johnstone, Dion Owens, designer
BULK SALES MANAGER
Ginnie Boudrieau
ASSISTANT TO THE PRESIDENT
Matthew Smith
ADVERTISING
603-924-7138
Louise Caron, John Gancarz, Susan Martin, Hal Stephens, Marcia Stone, Office Mgr.

APPLICATIONS

BUSINESS

EDUCATION

Page 63.

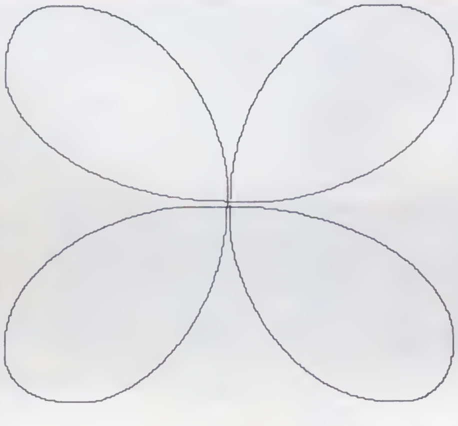


Page 76.

Contents: September 1981

Volume V
No. 9

Page 120.



Page 150.

This month:

Each month we invite our readers to tell us about themselves and their interests by answering the survey questions contained in the magazine. They are formulated to assist the editorial department in bringing you the types of columns and articles you want to read.

Each month we pour through the information received to get a better picture of who our audience is. This demographic information helps us to determine the types of articles that will appeal to the majority of our readers.

So, this month, turn to the perforated card facing page 210 and fill out the questions contained there. You can send in your responses confident that you are helping to shape the future of the magazine.

* * *

Have you ever been ripped off, or perhaps pleasantly surprised, by the type of service you received from one of our advertisers? *Kilobaud Microcomputing* welcomes your reports on the dealings—both good and bad—you've had with our advertisers.

Let us know if there were any problems in the delivery of your product—how many days, weeks, months? Was the product first-rate or poor? If repairs to the product were necessary, what about the customer service?

Be sure to state the name of the firm and the amount of the order. We're particularly interested in your mail-order experiences. Don't be sparing in your praise for the good firms, which should be encouraged. The bad ones should be stimulated to get their act together.

—The Editors

This month's cover:

Photo by Martin Paul. Special thanks to Dr. and Mrs. Robert MacCready and to the Jaffrey Historical Society for their assistance. Model: Cynthia Beth Nelson.

Kilobaud *Microcomputing* (ISSN 0192-4575) is published monthly by Wayne Green, Inc., 80 Pine St., Peterborough NH 03458. Subscription rates in U.S. are $25 for one year and $53 for three years. In Canada: $27 for one year only, U.S. funds. Foreign subscriptions (surface mail)—$35 for one year only, U.S. funds. Foreign air mail subscriptions —$62 for one year only, U.S. funds. Canadian Distributor: Micron Distributing, 409 Queen St. West, Toronto, Ontario, Canada M5V 2A5. In Europe, contact: Monika Nedela, Markstr. 3, D-7778 Markdorf, W. Germany. South African Distributor: KB Microcomputing, PO Box 782815, Sandton, South Africa 2146. Australian Distributor: Electronic Concepts, Attention: Rudi Hoess, 55 Clarence Street, Sidney 2000, Australia. Second-class postage paid at Peterborough NH 03458 and at additional mailing offices. Phone: 603-924-3873.

By Wayne Green

Instant Mail

Here's a Package That's Waiting To Be Delivered

The Billion Dollar Product

It's been a long time since I've written about a needed product: what I call the *Electronic Mailbox*. I'm not sure what is holding this up, but I do know that there is a good chance that the firm which gets it to the market first could end up selling millions of them and making billions of dollars.

The *Mailbox* will not only make its manufacturer a bundle, but will also have profound effects upon our whole field. It will totally change our system of business and personal communications. It will get small computers selling like nothing before has. It will make possible an array of data-supplying businesses.

The odd thing about the *Mailbox* is that there is nothing much to invent, since all it requires are the applications of already-discovered electronics. Indeed, most of the functions we need of it are being accomplished by currently available consumer devices. To be specific, we need a modem to connect our computer to the phone line. Well, we've had those for a long time. We need an automatic dialer, also a commonplace gadget these days. And that's about all we need besides some protocols and software, none of it very complicated.

You'll better understand what I have in mind if I define my *Mailbox* in terms of what it does. There have been so many articles on electronic mail that one can't help but be confused about the situation. I see ads from some computer firms saying that electronic mail is here. Horsefeathers.

The *Mailbox* should be so designed that it can plug into any of our microcomputers and work. A phone line would also be plugged in, with output available for either dial or button operation. I'd prefer to have the operating system on a ROM in the unit.

Here's how it would work. The operating system would include a word processor which would allow you to write messages or letters on your computer. The system would ask for the phone number of the addressee and whether you want the message sent immediately or at the low rate time (3 AM), with a default to immediate transmission. You want to have a chance to edit the letter and okay it for transmission. Once you hit "go," the *Mailbox* would take over and access the phone line. If you have a telephone on the same line it would disconnect this so you could not pick it up and screw up the message.

Next it would call the number. If the line is busy it would wait an appropriate time and try again. Upon connection it would send a signal tone which would actuate an identical *Mailbox* on the other end, disconnecting the phone and turning on the computer if it is off. More expensive models may have the facility for recording incoming messages without bothering the computer at all. This can be done with cassette or disk.

Once the *Mailbox* is on line and ready to receive the message, it would send a handshake signal. This would also be coded, to check out the line and make sure it is workable. My *Mailbox* would then dump my message, along with a count of the bits, so the receiving system can make sure the message was received correctly. We may work out error-correcting codes to help with this situation, if that turns out to be helpful.

When the message and the count are received and compared, the system would disconnect, providing that the message was okay. If not, the receiving *Mailbox* would request another try. Failing twice, perhaps the system should disconnect and try all over again. If this is a significant problem we'll solve it.

The receiving system would then have an indication that a message is awaiting, much like the light on a hotel room phone.

With telephone dialing taking 15–20 seconds, we should be able to make contact within a half-minute. I suggest that we use a 1200 bits per second (bps) data rate system to get started, planning on going to a much faster one later on—perhaps 9600 bps, which is being used with success over ordinary phone lines with a combination of compressors and expanders.

At 1200 bps, we are sending about 100 characters per second. This is about 16 words per second, or about 12 seconds for a 200 word page. Thus a complete dialing up, answering and exchange of message would usually be completed in less than a minute.

Before I go on to some further developments of this system which seem inevitable, let's just mull over what changes a one-minute message service could bring us. I hope it is no news to you that business letters are costing well over $7 each to handle these days—and take several days to do their expensive work. My *Mailbox* system can give you almost instant mail, and at a fraction of the cost of a dictated (or even a hand-typed) letter.

If you spend much business time on the phone, you will not be surprised to learn that recent studies have shown that 73 percent of the business calls do not go through on the first try. It's a busy line, or your person is out to lunch, golfing, messing around, on another call or in conference . . . choose one. And what percentage of call-backs do you get? It's low.

Most phone calls are for relatively simple matters which could be handled with an instant message. Longer discussions can be set up via the *Mailbox* for a phone call at a specific time, and acknowledged with another instant message.

How long will it be before we have more advanced *Mailbox* models which will automatically forward our messages to home, another office . . . and eventually to our car via a repeater system or a satellite? This could even reach us in an emergency via a pocket receiver *Mailbox*. Hams have no problem keeping in touch from just about anywhere in the country via a system of over 5000 automatic repeater stations. I can whip a hand-held

transceiver out of my coat pocket or from my belt and access the phone lines from almost anywhere in the country. Indeed, I can do this in about 50 countries around the world, so commercial use of this system should not be that far off.

What will happen to U.S. mail when it is up against an inexpensive and instant mail service? Well, they can always handle packages. No, they've turned most of that over to UPS. Well, there's always junk mail.

I believe that a system like this will provide the speed of communications which will make business much easier. It will allow more and more people to conduct most of their business from wherever they please, whether this be home, their yacht or a portable desk on the beach somewhere in the Caribbean.

Speeding It Up

Going to 9600 bps would be eight times as fast, getting a one-page letter out of the way in one and a half seconds. Not bad, but not as good as it could be, even with techniques we have today and nothing to invent.

The next step may be a bit confusing for neophytes to computers, but it is simple. Our standard byte has eight bits. Two of them use 16 bits. If we take one bit away, using it to indicate that the following 15 bits are a special code instead of two characters, we can have a dictionary of 32,000 words, each selected by a combination of those 15 bits. Thus, instead of having to send 6-1/2 characters (bytes) per word, we will only have to send two, a savings over the letter-by-letter method of 3-1/4 times. That would bring our time for sending a 200-word page down to under one-half second! In one minute, counting half of it for dialing the number, we could transfer 66 pages of text!

This would force Bell to change their billing system, for most calls would be a mere fraction of a second long. Further, the current system of starting billing after two or three seconds would not indicate any calls at all on your phone bill. I'm sure they would fix that situation.

In addition to increasing business and personal communications by an order of magnitude or two, this system would also allow a rapid expansion of automatic business communications. Any business wanting to keep track of sales could call up a remote system and get the data at any time. Chain stores could be quickly polled for the most detailed record of each day's sales, thus keeping inventories to an absolute minimum . . . and inventory costs. Hotel chains would know nightly how many rooms were used and the income therefrom. Businessmen could get daily accounting reports, if desired.

Since the dictionary would probably be on a ROM, it might be possible to develop interpreters to translate from one language to another. Thus I could type a message in English and it could be read out in German, Chinese or Hebrew. A switchable ROM interpreter would allow the output to be in any language you desire. Not a bad way to learn languages either.

Thus, with our simple *Mailbox* system we would enable an entire world to be able to communicate for the first time—all automatically.

Data Banks

The *Mailbox* would facilitate the use of data banks, thus encouraging their growth. It won't be long before we really won't need to have a large library of books on hand. We will be able to call a medical data bank and get a good guess on an illness . . . perhaps even to a remote taking of temperature, pulse, emotional reactions to computerized questions, all of which would render a probable diagnosis and uncover the underlying psychosomatic elements of the illness.

This is a good time to start laying the foundations for data banks. I believe that we will develop some very sophisticated ones, with information on any subject, incredible indexing, and both words and graphics exchange of material wanted. Sure, we'll be paying for the use of the data, but the time we save will make it well worth while. Oh, yes, those graphics will eventually be in full color.

Magazines of the future will be electronic and digital, as will most books. In that way the information will be available to even more people than today and publishers will make even more sales. Will magazines be monthly or come out a little bit at a time? We'll have to wait and see how that works out.

Bottom Line

All we are waiting for is some technician somewhere to put the package together, develop the needed software and get it on the market. If he is (or they are) shrewd he could end up like the Apple chaps with paper millions. If he screws it up, someone else will end up with the boodle. Whatever happens the microcomputer industry will get a boost it will never forget and we will be on our way towards a series of changes in society which will rival those brought on by the development of the automobile.

Would I fool you about something like this? Remember that I was writing about mass-produced software for a couple of years before anyone seriously got into the business; now it is a $100 million industry, aiming toward $25 billion in a few years.

If you do get busy with this, I expect to get some articles for *Microcomputing* about the system.

Why Software Firms Are Folding

We've had some fairly large losses recently in the software field, with no firms

In Language Lies The Future.

Paul Lutus creates language. Language for the future. *From him we have* Apple Writer, Apple World and MusiComp. His software was used by the Viking Mars Lander team and other advanced space programs.

For the past three years he has applied himself to the development of more efficient and powerful computer languages to help him with his work. These new languages of the future are now available for your use exclusively from Insoft.

Each package introductory price $125.00

See the new world of programming from Insoft at your Apple dealer.

↙274

insoft

The languages of the future. The tools of today.

259 Barnett Rd. / Unit 3 / Medford, Or 97501 / (503) 779-2465

Apple is a registered trademark of Apple Computer, Inc.

TransFORTH II

The final "word" in Apple programming. A brilliant extension of FORTH.

- Fully compiled
- Floating point
- Transcendental functions
- Strings and arrays
- Hires, Lores and Turtlegraphics
- Music

Far more compact and approachable than Pascal, TransFORTH II is both recursive and structured and easier to use than BASIC.

Over three years in development, this high level language enables you to program in English with far greater speed and convenience than ever before possible.

TransFORTH II. A transformation in computer programming. Available today.

ALD System II

The Assembly Language Development System. Already the preferred assembler for professional software developers.

- Object files to 18K
- Source files to 37K
- Cursor based screen editor
- Upper and lower case text entry
- Nested macro instructions
- Local, global and universal labels
- Comprehensive error trapping

The enormous file capacity of ALD System II assures you of the ability to develop virtually any imaginable software system. And you can do so with ease since ALD System II edits with the simplicity of a word processor.

ALD System II. Powerful yet convenient. The first choice of professionals.

The best comes last. Both new programs are available for the Apple II or Apple III.

really showing a strong posture. The Bottom Shelf went to the bottom, for example. Let's take a look at this situation and get some perspective on it.

With all of the promises of riches for mass-produced software, what is going wrong? Is it mainly copying of programs which is doing in the industry? Is it the weak financial situation of many computer stores which prevents them from carrying a large stock of programs? Is it the large amount of really awful software which has been sold and possibly disillusioned customers? Is it the pitiful software selection offered by Radio Shack?

All of these are factors. Let's look more closely at the situation and see what is wrong, and what might improve matters.

First, let's look at the major hardware firms, most of which have been providing a minimum of software support of their systems. Here, almost without exception, we have a hangover from the old IBM days. IBM wanted to sell systems which were totally within their control. They wanted no outside software or accessories. When the government brought a halt to this, IBM reluctantly went along with outside suppliers, but there is no hint that they like the situation.

When the minicomputers came along, they went the same route . . . with their own bus structures, their own languages and so on. Most of the major hardware firms are still playing that game.

The first microcomputer manufacturer, Mits, tried to set things up so they would be the sole supplier of all accessories, all software, and so on. They even wanted to have total control over their own stores, prohibiting them from selling anything not coming from Mits. They lost most of their good stores that way, paving the road towards their present oblivion.

Radio Shack is a good example of this mentality—an almost perfect example. No product not coming from Radio Shack is permitted in their stores. The store managers can't even give away non-Radio Shack products. I remember some early chats when I was considering a Data General computer and wanted to connect one of my many Teletype machines to it. They were absolutely adamant that no "hostile" equipment could be connected to their system. If it was I would get no service.

> A high percentage of the program packages being sold today are a pain for both dealers and customers.

This policy, which is pretty much industry-wide, has resulted in dealers carrying a minimum of programs for the systems they sell. Prospective customers going into a Radio Shack can well ask, "Is *that* all I can do with this?"

If a firm is going to supply much in the way of software for the major systems, they are going to find that they get little cooperation from the systems manufacturers. It turns out that it is extremely difficult to turn out cassette duplicates in any quantity that work reliably. With little, if any, help from the system manufacturer, and little outside data available, most software firms turn out bum copies and discourage both dealers and customers.

The duplicating system at Instant Software took over a year to develop and currently has some $50,000 or so invested in it. This is not as dependable as we like, so another $50,000 is scheduled to be put into this in the near future. We want to be able to get rid of the need for quality-control checking of programs which have been duplicated, yet at the same time be able to turn out tens of thousands of perfect duplicates in a day to keep up with the growing market.

Without an investment of that magnitude you have to make do with poor copies of programs. This is why a high percentage of the program packages being sold today are a pain for both dealers and customers. Some work . . . sometimes. Many don't, bringing aggravation

and expense to everyone involved.

Smaller software firms have to make do with junky looking packages. Some use stapled baggies on cardboard. We're finding that machinery to turn out good looking packages is in the $15,000 to $25,000 range, so without the quantities of Instant Software, packages would have to look cheap . . . thus not encouraging their sale in stores.

With over 2000 computer stores to service, and an equal number of small software firms wanting to supply them, the idea of doing business by mail gets ridiculous. No store wants to try and deal with 2000 suppliers—or even 50 software firms. The bookkeeping is impossible, even with a computer. I do suspect that computer stores will be the very last retail stores in the country to computerize.

The high prices of most software, brought on by the expenses of selling and advertising, encourage customers to make copies for friends. Theft. I find that many dealers are making this situation worse by wanting to get the maximum dollar profit from each sale, getting the money today and not worrying about the future. Some are even pushing Instant Software to increase prices so they can make more profit per sale. I think that is nonproductive in the long run.

If a firm is going to be successful in selling software, it is going to have to do most of the selling through computer stores, not by direct mail. This means that the firm is going to have to have a manufacturers' representative team which can get to every store at least once a month to check programs on display and refill the racks. To make this pay, the firm has to have enough software to make it worth the while of the reps to make these sales calls. Without some serious source of funds to build up to this size, software firms are in a Catch-22 situation. They can't get dealer sales without reps . . . and *they can't* get reps without a large number of packages to sell . . . yet without the sales they can't attract programs from programmers.

Instant Software solved that problem by investing some $3 million in the development of the people and plant needed to get the needed programs, to make sure of the quality of the programs, and have enough of them to make it possible for a rep to earn around $25,000 a year right from the start. Without the help of many other publishing functions, this could easily have escalated to $5 million or more, as other software firms have discovered.

The end result has been a concentration on mail order sales of programs instead of through dealers, plus the growing consumer resistance to new firms. This has helped to put many smaller firms out of business. The larger firms did not have enough programs to pay for reps, so they too languished, unable to either get enough orders or to get dependable payment from dealers.

As far as I know, only Instant Software is building a rep team to contact all of the computer stores and other distributors of small computer systems. Sales have been difficult because while the most popular system is the Radio Shack, the TRS owners have to buy their system one place and get the software for it elsewhere. This is a big benefit to the computer stores in that the TRS owners are thus forced to regularly go into non-Radio Shack stores and this tends to help sell Apples and other systems to this group. With Apple sales running about one-third those of the TRS, the market for Apple programs has been substantially smaller.

If we are talking mass distribution of programs, as soon as we finish talking about TRS and Apple, we are left with such small sales of hardware that it is no longer economically feasible to mass-produce and distribute programs through stores. Instant Software can do it, but only because the TRS and Apple sales pick up most of the costs of advertising and distribution. The other programs ride on the coattails of the better-selling systems. For the smaller firm there is no good answer to selling Heath, Commodore, Atari, TI and other programs . . . as yet.

Programmers are the main losers, of course. For having cast their lot with a firm unable to keep up with the competition, they generally lose everything.

The Solution

I am not alone in believing that the mass sale of microcomputers is tied to the mass sale of software. Thus, the more programs a system has available for the prospective customer, the greater the likelihood of a sale of the system. If the manufacturers of systems would understand this and react appropriately, sales would soon be rising and their advertising would become far more effective.

Radio Shack had better either get their system for obtaining software to sell in their stores into higher gear or else work out a way for supporting firms to sell through Radio Shack stores. With the coming barrage of new computers from IBM, Data General, DEC, Xerox and a variety of Japanese firms, Radio Shack can hardly afford the luxury of bungling their software support.

One other idea they are going to have to accept—a foreign one for them—has to do with their stores carrying *only* best-selling products. It is going to be necessary to carry a lot of slow-moving software as applications packages come out for smaller businesses. The factory may have to share in the inventory cost of slower-moving software, but it really has to be there if they are going to continue to grow in sales.

Other firms are going to have to be much more cooperative with software firms which can help them. They are either going to have to set up software evaluation departments or else deal with software suppliers whose reputation will give them the same results. Then they are going to have to do all they can to get this software into their dealer stores . . . again, possibly with some financial help for the stores or the software firms.

One other approach is to get sales started by providing the financing, at least in part, for the kickoff of new programs. This will get it fully developed, packaged and into the stores to prime the pump. From there on the software firm can continue to supply copies as the program sells. I suspect that a $1000 investment per program package would get an arrangement like this started. Thus, a $250,000 investment would assure 250-program support for the system. Dirt cheap.

With each new manufacturer of hardware anxious to invest the minimum possible—tiptoe into the water, as they put it—no one system is likely to achieve outstanding sales. The first firm that recognizes the importance of software support and puts up the money to supply it is going to have a tremendous advantage over the others. So far everyone has been lucky in that no firm seems to have figured this all out. Everyone but the software firms, which have been busy going broke.□

By Robert W. Baker

CBM Utilities

Get the Most Out of Your System

Filemaster

Here's a handy disk utility called Filemaster that the author refers to as a diskette management program. It was written specifically for the 8032, but separate versions are available for the 8050 and 4040 disks. The program offers a number of useful features, all selectable from a handy menu.

You can list or print a full diskette directory with 54 files per screen or 180 files per printed page. The Filemaster directory is the full diskette directory, including scratched files. For added convenience you can even alphabetize the directory, then later restore it to its original order.

The other functions operate on a list of files that's referred to as a job-cue. Special commands are included to add or remove files from the job-cue. Once defined, the job-cue can then be used with any number of commands, without having to be redefined each time. This is valuable to those producing software products consisting of several files. Now you can easily copy only the exact files wanted without having to type in separate copy commands for each file. You can create a specific job-cue for the product you want created and do the whole thing with only one or two commands.

Using the job-cue, you can scratch files or even recover previously scratched files (with limitations). Any file that cannot be fully recovered is partially recovered and properly terminated. Relative files can only be recovered as sequential files.

When copying files from drive 0 to drive 1, you can specify whether or not replacement is allowed. Another copy option will even scratch the files from drive 0 once they have been successfully copied to drive 1. However, I would recommend using caution when using this option.

Other features allow aborting the current job, checking and/or newing (formatting) a drive and paging the screen display. Cost of the program is $30, both versions for only $45. For more information you can write Software by Sasso, PO Box 969, Laguna Beach, CA 92651. I should mention that this was formerly California Software Associates, authors of several other fine disk utilities previously reviewed.

HES Products

Here are a few interesting and well-documented programs from Human Engineered Software of Los Angeles, CA. In fact, these are probably the best-documented programs I've seen for the PET/CBM. Each program comes with a complete description of the program operation itself, along with a variable list and cross reference. This is in addition to the normal description of the program operation. You can easily customize and enhance the programs. In fact, there are even suggestions of how you can make certain changes within the documentation.

HESlister

HESlister is a simple utility program that produces BASIC program listings in a formatted manner. It quickly shows the structure of your program and can even highlight possible bugs. Statements contained within a FOR-NEXT loop are indented, as are statements following an IF statement. Cursor control characters that appear within quotes as reverse characters are converted to two-letter codes within square brackets.

The program to be listed must be on disk, but the disk may be in either drive. Date and time are entered along with the program name. This information is printed at the top of each page generated. You can, however, direct the output to the screen instead of the printer. Once started, the program runs constantly until the listing is completed. There is no way to stop or suspend printing except via the stop button.

When using the program, I did come across one little problem. If you have a colon in a remark, the line is split as if it were separating BASIC statements. The colon in the remark is not really treated as a remark. Apparently the program just doesn't ignore data following a REM token to the end of a line. This shouldn't be anything to worry about and probably could be quickly fixed. Otherwise the program is very nice, but be prepared to use a little more paper. It's nice to have cursor controls in a form you can easily understand.

HESedit/HESbal

HESedit is a full screen text editor designed for creating and editing source programs for the HESbal assembler. It can also be used for upper/lowercase text editing. Since it is written in both BASIC and machine code, separate versions are required depending on your machine. Instructions are also included for interfacing disks and printers.

HESbal is supplied as a one-pass assembler for an 8K PET/CBM with a single cassette. It can easily be made into a two-pass assembler with a few simple program modifications if you have a disk. The program is actually written in BASIC, so it should run on any ROM version. On the other hand, because it is written in BASIC, it is much slower than higher-priced assemblers *written in machine* code.

This assembler reads the program source file from cassette tape as generated by the HESedit program. The assembled machine-code program is generated directly in RAM memory in either the second cassette buffer or upper memory. You are limited to about 1000 bytes in an 8K PET.

Statement labels can be one to six characters long. Op codes are standard three-letter 6502 mnemonics. All 12 addressing modes are accepted, as well as decimal and hexadecimal constants. However, since this is a one-pass assembler, symbols must be defined prior to being used (no forward references). One nice feature is the inclusion of extensive error reporting.

HESedit and HESbal are normally pro-

vided together as an assembler package, but a copy of Micromon is also included. This is an enhanced version of Extramon written by Bill Seiler. However, Micromon will only run with BASIC 3.0 or 4.0 ROMs and has a few restrictions.

HESlister sells for $12.95, while the assembler package is $23.95. Postage and handling is an additional $1.50. For more information you can write Human Engineered Software, 3748 Inglewood Blvd., Room 11, Los Angeles, CA 90066. Future enhancements and additional items were planned at the time I received the sample copies.

UARCO

UARCO, Inc., one of the largest manufacturers of business forms and systems, has chosen the Commodore CBM system as the featured system in its new supplies catalog. The catalog also offers a collection of software packages, including VisiCalc and other business and word processing programs. This should make the CBM one of the more visible systems in the business environment.

A recent Commodore press release quoted UARCO's general manager of direct mail marketing as saying they chose Commodore "because their small, desktop computer was by far the most powerful and most reliable system on the market in a price range that is considered affordable by a small business." Although customers can buy the system directly from the catalog, the company has a large nationwide sales force that can arrange for support if needed.

As for service, UARCO has service centers throughout the country for on-site repairs for their Commodore customers. Loaner systems are available if needed. A toll-free hotline is also available for information and operator assistance. It's a shame we didn't get this kind of support when the Commodore PET was first introduced!

CHRGET/CHRGOT Routines

Many people have dabbled in using the various machine-language routines contained in the PET/CBM operating system. But don't forget there's one very useful routine in lower RAM memory.

When the system is first powered on, a small 24-byte subroutine is written in lower RAM memory during the system initialization. This routine is normally referred to as the CHRGET (or CHRGOT) routine. It's generally used to get the next character or BASIC token from a BASIC program line stored in memory.

When the routine is called, any spaces are automatically skipped and the 6502's carry bit is set to indicate the type of character found. If the character is numeric, the carry bit will be cleared (0). Otherwise, the carry bit will be set (1) for all

```
N:DISASSEMBLY OF CHRGET/CHRGOT
N:               RAM SUBROUTINES
N:               (BASIC 3.0)
N:
D:0070-0087   1  2  3   MNC-CODE
I:0070       E6 77      INC $77
I:0072       D0 02      BNE $0076
I:0074       E6 78      INC $78
I:0076       AD 06 02   LDA $0206
I:0079       C9 3A      CMP =$3A
I:007B       B0 0A      BCS $0087
I:007D       C9 20      CMP =$20
I:007F       F0 EF      BEQ $0070
I:0081       38         SEC
I:0082       E9 30      SBC =$30
I:0084       38         SEC
I:0085       E9 D0      SBC =$D0
I:0087       60         RTS
N:
N:
N:********** HEX DUMP **************
N:
N:
M:0070-0087   0  1  2  3  4  5  6  7
W:0070-0077  E6 77 D0 02 E6 78 AD 06
W:0078-007F  02 C9 3A B0 0A C9 20 F0
W:0080-0087  EF 38 E9 30 38 E9 D0 60
```

Program listing.

non-numeric characters and BASIC tokens. In either case, the 6502's Z-bit will be clear (0) unless the character was a colon (:). The character itself is always returned in the 6502's accumulator.

I've included a simple disassembly listing that shows the routine residing from $70 to $87 (112–135 decimal). Whenever a JSR $0070 is executed, the address pointer stored within the LDA instruction (at $76) is first incremented and then the character is fetched and tested. This is important; remember that the address is incremented first, before the character is fetched. Note that by doing a JSR $0076 a program can refetch the last character without disturbing the address pointer. Also, the two SBC instructions ($82 and $85) effectively set the carry bit as appropriate to indicate the character type without actually changing the character in the accumulator.

You can readily see that this is a handy routine that is used heavily by BASIC itself. However, there are a few ways that you can use this routine for your own programs as well.

One idea that many people have used successfully is to replace the first three bytes of the CHRGET routine with a JMP instruction. The JMP instruction then goes to another user-written machine-language routine to check for some specially-desired action. The routine does its thing and then returns to the CHRGET routine. The added routine is always executed whenever BASIC fetches a character from a program line.

This is one simple way to add new BASIC tokens or implement a crude program to trace execution of your BASIC program. When you see this technique you must remember, however, to execute the CHRGET instructions replaced by the JMP instruction before returning to the remainder of the routine.

If you are writing a machine-language routine, you can use the CHRGET rou-

tine directly to read data from anywhere in memory. First save the current address pointer (from $77 and $78) somewhere safe in memory. Next set your desired pointer (less one) in the same locations. Remember to use correct 6502 address formats (low byte first, high byte last). Then simply call CHRGET (and/or CHRGOT) as required. When done don't forget to restore the original address pointer that you previously saved before returning to BASIC.

If you are writing a stand-alone machine-language program, you don't have to worry about saving and restoring the address pointer, since BASIC isn't being called. Just set the pointer as needed and call the routine.

Before I forget, here's another possible use of the CHRGET routine that I haven't seen anyone really talk about. This is an easy way to pass parameters to machine-language routines. It is somewhat limited since you cannot pass parameters from BASIC variables, but it may be useful for certain applications.

Remember that the CHRGET routine is always used to read each BASIC line that is being executed. If you execute a SYS(. . .) . . . . command the address pointer in CHRGET ($77 and $78) will be pointing to the colon following the SYS command. If you have a REM statement

Commodore now has a Software Encyclopedia available through most dealers.

following the SYS command you can use the CHRGET address pointer to know where to read any data from within the REM. If your program reads the data directly without using the CHRGET routine, then BASIC will simply skip over the REM when your routine returns to BASIC.

However, it's even easier to use the CHRGET routine to read SYS(. . .) call parameters. You still need the colon after the SYS(. . .) to terminate that statement correctly. On the other hand, you no longer need the REM since calling CHRGET will automatically increment the address pointer past the parameters read. If you read the correct number of characters before returning to BASIC everything will continue as normal. To protect yourself you'll probably want to terminate the parameters with a colon as normal. In your routine, then, you just continue reading data until the colon is found. Remember that the 6502's Z-bit will be set when the colon is found so it's very simple to check for.

Thus, you could use something like:

```
SYS (12345): "THIS IS A TEXT HEADING":
X=: . . .
```

to pass a text string heading to your machine-language routine.

As I said before, this method is somewhat limited, since you cannot pass the value of a BASIC variable. Any parameters to be passed must be actually written within the program.

Miscellaneous

I finally got hold of a manual for Commodore's new 4022 printer. I couldn't get a printer so at least they sent a manual. Reading the manual, it appears to have the same features as the familiar 2022 printer. It offers the same controls with just a few limitations here and there. You can still do formatting, expanded printing, special characters, etc. The earlier models only had single-direction printing, but current models have bidirectional printing. One thing you must observe with the newer printers is to allow a cool-down time of at least 998 microseconds for each line of reverse field after six continuous lines of reverse field. All in all, the new models look nice.

Commodore now has a Software Encyclopedia available through most dealers at a reasonable cost. It lists over 400 programs in seven categories: business, word processors, utilities, engineering aids, personal aids, games and education. The book also lists 51 software vendors complete with addresses and phone numbers.

Listing of a particular product does not indicate an endorsement of the product by Commodore unless the Commodore logo appears with the listing. This list is by no means complete as there are many products currently advertised that are not included. It is not nearly as comprehensive as Robert Purser's former Cassette review, which listed many more Commodore programs. However, it is the only such list currently available showing only Commodore programs. Hopefully, future editions will be more comprehensive as more vendors become aware of its existence.

I thought I had better include a short note on my own personal happenings to let people know what is going on here. At the time this column is being written, I have started a new full-time job and am doing a fair amount of traveling. I've shut down many of my outside activites but intend to keep writing this column and other miscellaneous articles as long as my time permits. However, there is the possibility that I will be relocating to the Midwest sometime in the next six to 18 months. Also, I will definitely be doing some traveling from time to time. If you write with a question or comment and I don't get back to you right away, please hang in there—I will get back to you as soon as possible. Also, don't forget to include an SASE if you expect a reply. These tend to get answered first.□

Address correspondence to Robert W. Baker, 15 Windsor Drive, Atco, NJ 08004.

✓See List of Advertisers on page 210

By Frank J. Derfler, Jr.

Tapping The Brain

Is it Fact or Fiction?

In this month's Dial-up Directory, we're going to look at some science fiction and some science fact. The fiction is a great story that all readers of this column will enjoy. The fact is a review of a very nice pair of communications programs for the Apple II.

A Book Review

Dell Books publishes a paperback series called Binary Star. Each one contains two science fiction stories—one by a known author and one by an unknown. Binary Star #5 carries a story called "True Names" by an unknown author named Vernor Vinge.

The story is set in a time some years in the future when data communications and information networks have reached into every home and every aspect of life. The man-machine barrier has been broken, and those who can afford it have interface devices that can enter data into and take data from the brain without going through the narrow communications ports of the hands, eyes and ears.

In Vinge's story, the brain doesn't display neat strings of ASCII characters inside people's eyeballs. Rather, the information transfer is done in the form of images very much like daydreams. Many people use mystical or storybook imagery such as we find in the Adventure or Dungeons and Dragons games to communicate. Persons operating on this "other plane" of interface with the world send and receive data in the form of symbols. Those "chatting" together (as on The Source or CompuServe today) can interact through the images they assume.

Then, as now, many groups exist which enjoy breaking into new databases for fun and profit. The most successful of these data wizards join their images in a secret coven which meets in a castle, megabytes in size, which is guarded by a dragon in a moat of lava.

The best-guarded secret of a wizard is his or her "true name" (entry code, password, etc.). Anyone knowing a wizard's true name can become a blackmailer by threatening to destroy images and even change access schemes to lock the wizard out of the other plane—very nasty business.

The true name of the hero of the story is discovered by the most evil of forces: the Feds. The Feds want our hero to fink on his friends, but suddenly a much greater danger threatens and our hero and heroine join together to wage a titanic struggle against evil within the data communications networks of the world.

I have written some fiction, which is tough to do well. Vinge weaves a good story with many surprises. But those of you who read this column will enjoy the technical accuracy, realism and tremendous potential of the data communications world Vinge creates. Your next session on the local CBBS will seem pretty tame after you read "True Names"! It lists for $2.50 at any bookstore.

Breaking the Barrier

The man-machine barrier must fall. I am sure it will, sooner than any of us expects. When we find a way to interface into the brain's main data bus, we will truly know what communications is all about.

This interface may take several forms. Initially, physiological means such as reading the impulses to small muscles may be used. Our keyboard entries are little more than muscle twitches of the fingers controlled by the brain. Fairly simple electrical sensors monitoring muscles in the extremities may prove to be a better way to output data from the brain with less conscious thought than typing. It may be possible to input data into the human system through electrical stimulation.

Analog Magazine had a fascinating series of reports about the transmission of sound directly to the brain through the use of modulated low-frequency radio waves. Unfortunately, being bathed in strong rf fields all day probably isn't too good for the rest of your biological system (although it happens to thousands daily). But some more direct means of data transfer must be found. Reading, listening, typing and talking are just too slow. Vinge is probably right when he describes systems using the brain's own dreamy symbology. You creative biologists/computerists out there, get moving!

Postscript to the review of Binary Star #5: The "known" author, George R. R. Martin, was allowed to write an afterword comparing the two stories. He stated that his story, centering on a faster-than-light cruise ship and psychic powers, was "hard science fiction" compared to Vinge's. Ten years from now, I hope Martin is still trying to receive his mail by ESP. Vinge is the prophet.

Return to Today

I can feel the red pencil of my editor creeping up on me. He wants me to write something about data communications today. The best I can do is introduce you to two excellent data communications programs for the Apple II computer written by one of the developers of the ABBS system, Craig Vaughan.

Craig is marketing a complete series of Apple II programs (text editor, mailing list, message system, etc.) through his company, Software Sorcery. I doubt if there are many around who know more about programming for the Apple II, and all of the programs offer excellent value for the money. He has several terminal programs, including two running under Apple/UCSD Pascal, but we will look at two hard-working programs for the standard Apple BASIC Disk Operating System.

BITS and Hyper-BITS

BITS stands for Basic Interactive Terminal Software. It is a workhorse terminal program selling for only $44.95. Hyper-BITS performs essentially the same functions, only it uses interrupts to allow communications up to at least 4800 bits per second (bps). It sells for $64.95. Either program will run on a 32K Apple II with a disk and Applesoft in ROM or a language card.

Both programs are written so that they

Frank J. Derfler, Jr., PO Box 691, Herndon, VA 22070.

Beat The Socks Off Your Computer's Quarterback— You Hope
Tuesday Morning Quarterback
EPYX
...You've had a rough day. But now it's fun time; you've just loaded the brand new Tuesday Morning Quarterback™ program into your computer. You're the quarterback. And the coach, general manager, star. You're going to beat the socks off your computer. Or off a human opponent. You hope.
You pick the NFL team that plays closest to your management style. Now you assign a team to your computer ... or let a friend pick out an opposing team. If you want a better chance to win, pick a strong team; give your computer a weak team. (Don't try giving a weak team to a friend; just make sure you get first choice. After all, it's your game.)
You still have to play smart, though. And have a bit of luck on your side, too. Otherwise, your computer might beat the socks off you. (Later, as you get better, you can show your smarts by having your weaker team beat your computer's stronger team.
Now it's game time. List your line-up. If it's your ball, call the plays: running plays like quarterback sneak*, draw*, sweep* ... 6 choices. Or pass plays: short curl*, screen*, bomb* ...8 choices. And option* and trick plays, too.
The opponent has the ball. Call a normal 4-3 or 3-4 defense*. Or pull a line-backer out for a nickel* pass defense. Or call for the blitz*; you figure the other team is going to pass and you'd like to sack* their quarterback. But if their team elects to run, you're probably in trouble. 6 defensive strategies, including the Butkus*.
Tuesday Morning Quarterback. 16 offensive plays; 6 defensive strategies. And rosters to pick from, draft choices and trades to make. With a probability structure that comes unbelievably close to the real thing. It's never the same game twice. Superb graphics, real-time. Even color (with the Apple) and sound. You won't get bored playing the game. Not in your or your computer's lifetime.
Another bug-free, easy-loading lifetime computer game from EPYX. With the unique EPYX lifetime warranty: If anything happens to your cassette or disk at any time and for any reason, send it back with just $5.00 for shipping and handling and we'll send you a brand new one.
(Of course, there is also our 30-day unconditional guarantee: if your EPYX game has any defect whatsoever within 30 days of purchase, return it to us or to your dealer and we will replace it free.)
Visit your dealer now and pick up Tuesday Morning Quarterback in its good-looking, protective box with the best instruction book you ever read. Now available on disk for the Apple II (48K RAM with Applesoft) and Radio Shack's TRS 80 (48K RAM). Only $29.95.
If your dealer is out of stock and you can't wait, order directly from Automated Simulations. $29.95 plus $2.00 for shipping & handling (and sales tax if you're in California). We guarantee you will be delighted with Tuesday Morning Quarterback. (If it isn't exactly what you want, return it to us for full refund. No questions asked.) Enclose your check. Or if you order by Visa or MasterCard, use our toll-free phones: In the United States: operator 861 (800) 824-7888; In California: operator 861 (800) 852-7777; In Hawaii and Alaska: operator 861 (800) 824-7919.
Order today. You and your computer deserve the fun.
*These terms are all explained fully—and delightfully—in the glossary of the big instruction book ... a book so good even a novice will soon understand pro football. And enjoy.
AUTOMATED SIMULATIONS, INC.
Dept. TQ8, P.O. Box 4247
1988 Leghorn Street
Mountain View, CA 94040
55
EPYX
COMPUTER GAMES THINKERS PLAY

are easy to install and use. Novices in data communications should have no problem installing the software and using it to save data received from information utilities and message systems and to transmit prepared messages and files. This ease of installation is particularly important because so many different display boards, serial boards and modem options are available for the Apple II.

Each copy of BITS includes a configuration program which asks the user a series of simply phrased questions about the system it is to operate in. After the questions are answered, the configuration program custom-tailors the terminal software. You can't fully appreciate the value of this kind of service if you've never tried to personalize a machine-language program that must do as much I/O work as a terminal program does. You can use 80-column display boards, the Hayes Micromodem II, various printer ports or any one of several serial cards. BITS will configure itself to meet your needs.

The BITS program doesn't appear to be frilly. It uses the simple menu shown in Table 1 to help the user select the desired functions. However, a lot of activity is going on behind the scenes. Options A and B shown in this menu are only operative with the Hayes Micromodem II. If you don't have a Micromodem II, they disappear and option C becomes A. Options D, G, H and K toggle between two conditions (off/on, etc.), and the currently selected option is always shown. Submenus provide for additional choices such as transmitting files in prompted form, letter by letter, line by line or in a big long dump.

BITS has a number of options I like. When the buffer is full, you don't have to save it as a file if you don't want to. (Several programs on the market offer you no choice—when the buffer is full, you have to name a file and save it. My disks are full of files named "JUNK.")

The program allows you to read the disk directory before you save a file (again, unlike many other programs on the market). The control codes are easy to send, and one key can be defined by the user to transmit any ASCII character. This is very handy if you are active on CompuServe because you can define the key to be that right bracket you need for file commands which otherwise isn't available.

The program will store any command string you may want to use to initialize your printer. If you have an Epson MX-80, you may want it to print your on-line copy double-spaced or in compressed format. Normally, you would probably have to remember to send it the proper commands before going to the terminal program. BITS will send the printer whatever command you prestore in it to initialize its operation.

There are only two changes I would like to see made. I would like to be able to open and close input to the buffer from the on-line mode without going back to the menu, and I would like to be able to read the buffer without having to call it back out as a disk file. Craig responds to these nit-picks by pointing out that it is a $45 program, not a $150 one. Honestly, I don't know of a program for *any* system which has both of these features, but I keep on trying.

Faster, Faster!

BITS is perfect for the most common data communications work up to 300 bps. A growing number of people, however, want to communicate faster using the Bell 202, 212 and other signaling schemes or over direct wire connection to local systems. Craig's Hyper-BITS program can effectively move data at any rate up to at least 4800 bps. It can use several different serial boards including the Communications Card from Apple Computers, the AIO card from SSM and the California Computer Systems Asynchronous Card.

All of these interface cards need to be altered to allow interrupts to be generated by them. This is the only way to make the software fast enough to operate at high rates. The modifications are all minor and well documented in the Hyper-BITS manual. (The Hayes Micromodem II will not go over 300 bps, but it will work with this program at that speed.)

The interrupts essentially call for the attention of the CPU. Usually, the CPU operates in a polling scheme where it looks for a character from the keyboard, looks for a character from the serial port, etc. If it is dealing with a keyboard character, it can miss a high-speed input from the serial port. A generated interrupt says, "Look here!" A small ring storage buffer serves to hold characters in case the CPU is otherwise occupied.

Both BITS and Hyper-BITS come with several utility programs. Three programs are provided to create text files from operating programs: one for integer BASIC, one for APPLESOFT and one for machine-language programs. FILE PRINT will allow the user to read a file. Craig also includes a public file which lists all of the known ABBSes in the world. This file can be used to autodial the Micromodem II. A user-prepared file can do the same thing on a more personal basis.

Interestingly, during initial installation, the configuration program asks what telephone area code you are in. After that, if you autodial a number inside your area code from the public list, the area code isn't used. Outside of your local area, the code is used.

Both programs come with manuals which answer the most common operating questions and give information on the utility programs. No index is provided.

Programs should be simple to use. The software should work hard so the user doesn't have to. Craig Vaughan's terminal programs operate in exactly that way.

Check your local computer store or contact Software Sorcery, Inc., 7927 Jones Branch Drive, Suite 400, McLean, VA 22102. Phone is 703-385-2944. Craig's great developmental ABBS #1 is at 703-255-2192.

CQ DX Europe

In amateur radio, DX stands for long-distance communications. We have published several phone numbers for Canadian message systems, but there seems to be only limited activity in other areas.

Peter Goldman lives in Surbiton, Surrey, which is just outside of London. He is running a message system there, but he reports the rest of Europe has little activity. He says there are two Forum 80s in England, but they keep irregular hours. There is supposed to be a Forum 80 in Holland, but Peter reports no luck in getting up on it. I will gather some data on European systems and provide a report soon.

Remember, any European systems you may see listed probably operate on the European tone standards, which absolutely are not the same as Bell 103, so don't waste a phone call if you are not equipped to handle the European signaling scheme.

Never Know . . .

You never know what is going to show up here in Dial-up Directory. We feature reviews, news and previews. The responses from the readers continue to grow in number, so we must be doing something right. If you have any products for data communications you think our readers would like to hear about, drop me a line at PO Box 691, Herndon, VA 22070. Send electronic mail to TCB967 on The Source, 70003,455 on CompuServe or to the AMRAD CBBS 703-734-1387.□

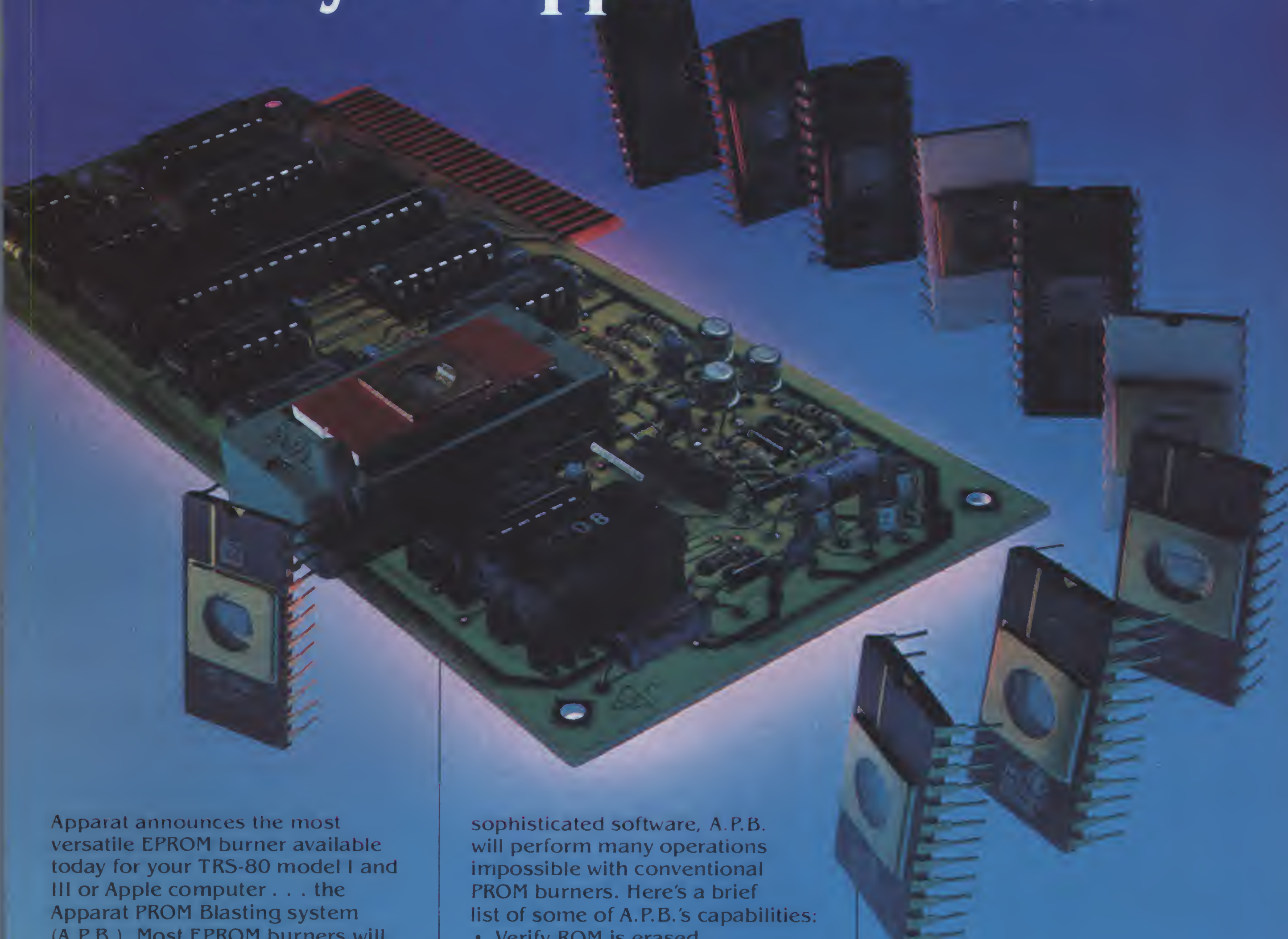
A. Exit terminal mode—stay on line
B. Exit terminal mode—hang up phone
C. Dial up remote system
D. Turn copy (on)
E. Save buffer to disk
F. Transfer file to remote system
G. Turn (off) line formatting
H. Set (half) duplex mode
I. Clear buffer
J. Print buffer
K. Turn printer (on)

Control B—send break
Escape—send escape

Your choice?

Table 1. BITS command menu.

Take a look at our EPROM blaster for your Apple™ or TRS-80.™



Apparat announces the most versatile EPROM burner available today for your TRS-80 model I and III or Apple computer . . . the Apparat PROM Blasting system (A.P.B.). Most EPROM burners will program only one type of EPROM. The A.P.B. system will program all commonly used 24 pin EPROMs by using special personality modules that adapt the unit to the EPROM. The following EPROMs are programmable: 2704, 2708, 2716, (3-volt) & (5-volt), 2732, 2508, 2516 and 2532.

The versatility and power of the A.P.B. system means you're getting a PROM burning package with extensive capabilities. The price, $149.00, insures you're getting the most cost-effective PROM burner on the market today.

With a unique combination of personality modules and sophisticated software, A.P.B. will perform many operations impossible with conventional PROM burners. Here's a brief list of some of A.P.B.'s capabilities:

- Verify ROM is erased
- Read ROM
- Copy ROM
- Copy between different ROM types
- Program ROM
- Partial programming and copies
- Verify programming
- Read or save ROM data on disk or cassette (Apple only)
- Program directly from computer memory
- Examine and/or modify working memory
- Preset working memory

The A.P.B. system is the most flexible PROM burner available. The A.P.B. system package consists of an interface card that plugs directly into an expansion slot*, a complete set of personality modules, software on disk and a detailed instruction manual. The software will run under NEWDOS/80, or TRSDOS™ for the TRS-80, and APPLE DOS™ or APEX for the Apple.

If you're looking for a powerful, versatile and cost efficient EPROM burner, call APPARAT today. Dealer inquiries welcome.

(303) 741-1778

*TRS-80 version requires the TRS-80 bus extender (Cat. # 1-025) or a separate user provided power supply and cable.

TRS-80, and Apple are trademarks of Tandy Corp. and Apple Computer.

VISA master charge

Apparat, Inc.

287

4401 So. Tamarac Parkway, Denver, CO 80237 (303) 741-1778

"ON GOING SUPPORT FOR MICROCOMPUTERS"

By Walter Koetke

Classroom Graphics

Do They Improve Instruction?

One of the more appealing capabilities of the microcomputer is providing the user with impressive graphics displays. The ease with which elaborate graphics displays can be programmed is amazing when one looks back just a very few years. The Apple II, the TRS-80 Model III and several other machines allow junior high school students to create graphics displays that would have certainly been legitimate topics for a master's thesis five or six years ago. What was the cost of the first animated PLATO displays? They were produced using expensive equipment by creative, talented and expensive professionals.

Similar and even better displays can now be produced on microcomputers in the $1000–$3000 price range by creative, talented students who are several years away from high school graduation. The young whizzes such as Bill Budge who've already graduated are doing even more impressive work with graphics. Could this be a reason for obtaining a high school diploma?

Does the use of color enhance instructional applications of the computer? Those resisting computer use and those selling microcomputers without color capability will quickly point out that no research has yet proved that color improves instruction. In my opinion, their logic is as sound as those who claim that no research has yet proved that smoking affects your health. Given the choice between using a microcomputer with a color display and one with a black and white display, I rarely see students select the latter. If the program the students are using doesn't use color, then the creators of that program have ignored a useful instructional tool. Are you listening, MECC?

Does the use of graphics enhance instructional applications of the computer? The last paragraph should be read again substituting "graphics" for "color."

If you observe students running programs they elect to use, you will almost always see many graphics displays. Although the MECC programs lack color, many MECC programs make exemplary use of Apple II graphics.

Graphic Examples

Many introduction-to-BASIC-programming texts reserve any mention of graphics for the chapter on advanced techniques. Many introductory courses never mention the word. This is an unfortunate situation, probably based on the historical difficulty of dealing with graphics. Using graphics is now easy and fun. Following are several examples that can be used when teaching students with very little experience using BASIC and no experience programming graphics displays. All programming examples are given for both the TRS-80 and the Apple II.

Our first graphics task will be to draw a rectangular frame around the screen on the TRS-80. The program can be used to introduce PRINT@ in the CHR$ function if students aren't already familiar with them. The TRS-80 can display characters in 64 columns and 16 rows. Each character position can be fully identified by an integer 0 through 1023. The first row contains positions 0 through 63 (from left to right), the second row 64 through 127, the third row 128 through 191 and so forth. Radio Shack's BASIC manuals contain a "Video Display Worksheet" that clearly labels each of these positions.

Before continuing, be sure you understand why the following program will print HELLO in the middle of the screen and READY at the bottom:

```
10 CLS
20 PRINT @ 448 + 29, "HELLO" ;
30 PRINT @ 896, ;
40 END
```

Radio Shack's manuals also contain a page showing all of the available graphics characters and their ASCII code. For our frame we shall need the three characters illustrated in Fig. 1.

Writing the program is then done in a rather straightforward manner. To clear the screen and display the 64 characters that make the top of the frame we use:

```
10 CLS
20 FOR P=0 to 63 : PRINT@ P, CHR$(131); :
   NEXT P
```

Next we add the right side, moving from top to bottom using:

```
30 FOR P=63 to 895 STEP 64 : PRINT@ P,
   CHR$(191); : NEXT P
```

Then we add the bottom (from right to left) and the left side (from bottom to top) using

```
40 FOR P=894 to 832 STEP -1: PRINT@ P,
   CHR$(176); : NEXT P
50 FOR P=832 to 0 STEP -64 : PRINT@ P,
   CHR$(191); : NEXT P
```

And finally we end gracefully without disturbing the rectangle using the commands:

```
60 PRINT @ 896,;
70 END
```

Before you write letters telling me of shorter programs to do the same thing—for example, replace all of line 20 with PRINT@ 0, STRING$(64,131);. . .—remember the suggested purpose of this exercise. We want to introduce graphics to the beginner. Doing this with a pocketful of special functions and clever tricks rather than with already familiar commands is not a sound educational practice.

Now that the TRS-80 is proudly displaying your rectangle, let's write the same program on the Apple using Applesoft BASIC. We'll do all of our displays using the Apple's low-resolution graphics screen. If students aren't familiar with the Apple's graphics mode, this program can be used as an introduction. Using this low-resolution mode, the screen should be thought of as containing two sections.

The upper graphics section contains 40 rows and 40 columns. The rows are numbered 0 to 39 from left to right, and the columns are numbered 0 to 39 from top to bottom. Each of these 1600 positions can be used to display a small colored rectangle. There is no selection of graphics characters as on the TRS-80; a rectangle is your only choice.

The lower text section can display four lines of conventional text in the usual manner. Note that in low-resolution

mode, this is the only section of the screen on which you can display any text. You cannot, for example, display the word "HELLO" in the middle of the screen while in this mode without writing a program to construct each letter from the displayable rectangles.

Our Apple program begins by switching to the low-resolution mode, clearing the screen and then picking a color.

```
10 GR
20 HOME
30 COLOR = 13
```

We can now proceed in a manner quite similar to that used in our TRS-80 program. The variable C will represent the column, and R, the row of the point being displayed. The top of the frame is created using:

```
40 R = 0 : FOR C = 0 to 39 : PLOT C,R : NEXT R
```

If students aren't already familiar with PLOT, this program provides a good introduction. The right side of the frame is then displayed using:

```
50 C = 39 : FOR R = 0 TO 39 : PLOT C,R :
   NEXT R
```

And finally, the bottom and left side are completed using:

```
60 R = 39 : FOR C = 39 TO 0 STEP - 1 : PLOT
   C,R : NEXT C
70 C = 0 : FOR R = 39 TO 0 STEP - 1 : PLOT
   C,R : NEXT R
80 END
```

Since the Apple contains the HLIN and VLIN commands to display horizontal and vertical lines and this program does exactly that, students should be shown the following alternate form for lines 40 through 70.

```
40 HLIN 0,39 AT 0
50 VLIN 0,39 AT 39
60 HLIN 39,0 AT 39
70 VLIN 39,0 AT 0
```

As a second example, let's look at a program that will randomly plot points until interrupted by the user. We might title this "Heavy Snowfall." Again we'll begin with the TRS-80.

The smallest point we can plot on the TRS-80 is a rectangle that occupies one-sixth the space occupied by a single character. A location is specified by indicating the appropriate row and column, much like the method used on the Apple. There are 128 columns numbered 0 to 127 from left to right and 48 rows numbered 0 to 47 from top to bottom. Specific rectangles can be turned on and off using SET and RESET, respectively. Our TRS-80 random snowstorm can be produced by selecting a random column and row, then turning on the rectangle that occupies that position. The same procedure is then repeated many times. A program that does this is:

```
10 CLS
20 C = RND(128) - 1
30 R = RND(48) - 1
40 SET(C,R)
50 GOTO 20
60 END
```

Simple, isn't it? You'll be amazed at the number of people who will watch the display produced by this program, and then wish to watch it again. Let's make the display somewhat more interesting. Rather than continuously plotting points until the screen is eventually all white, we will add a single condition to our program. If our randomly selected point is off, we will turn it on just as before. However, if the randomly selected point is already on, then we will turn it off. This is easily done by introducing the commands RESET and POINT. The only modification required is a new line 40, which reads:

```
40 IF POINT (C,R) = - 1 THEN RESET(C,R)
   ELSE SET (C,R)
```

Our modified program will fill about half of the screen. When this occurs the number of points being turned on will be about equal to the number of points being turned off. The result is a display that twinkles as it slowly changes form. Could this be a primitive form of computer art?

Let's now move our snowstorm to the Apple. The most notable difference is Applesoft's use of the more common, unwieldy form of the random number generator. An initial Apple program that will eventually fill the screen is:

```
10 GR
20 HOME
30 COLOR = 15
40 C = INT(40*RND(1))
50 R = INT(40*RND(1))
60 PLOT C,R
70 GOTO 40
80 END
```

This time the modifications to turn off an already turned on point are a little more tedious, since there is no Apple command similar to RESET. These changes are:

```
55 IF SCRN(C,R) = 15 THEN COLOR = 0
70 GOTO 30
```

Don't read on if you don't understand. Unless you've programmed the Apple or read about low-resolution graphics, line 55 will not be immediately clear. The idea is that a point is turned off by turning it on using the color black. A little cumbersome, but it certainly works. If you'd like a random display of colors, just change line 30 to read:

```
K = INT(15*RND(1)) + 1 : COLOR = K
```

and use K instead of 15 in line 55. Using colors, however, requires that you think of another name for the display.

As a final example, let's display a bouncing rectangle. Try doing this on your own before you continue reading.

To write this program students must know the concept behind animation. The illusion of motion is created by turning a point on and off, then turning an adjacent point on and off, and so forth. A bouncing ball on the TRS-80 can be displayed by:

```
 10 CLS
 20 FOR R = 47 TO 0 STEP - 1
 30 SET (64,R)
 40 RESET (64,R)
 50 NEXT R
 60 FOR R = 0 TO 47
 70 SET (64,R)
 80 RESET (64,R)
 90 NEXT R
100 GOTO 20
110 END
```

Do you understand? If you do, then you should be able to make the ball bounce faster. You should also be able to make it bounce slower. Can you add the effects of gravity so that the heights of consecutive bounces are reduced until the ball stops?

One Apple program that bounces our ball until stopped by gravity is:

```
 10 GR : HOME
 20 FOR H = 0 TO 39
 30 FOR R = 39 TO H STEP - 1
 40 COLOR = 13 : PLOT 20,R
 50 COLOR = 0 : PLOT 20,R
 60 NEXT R
 70 FOR R = H TO 39
 80 COLOR = 13 : PLOT 20,R
 90 COLOR = 0 : PLOT 20,R
100 NEXT R
110 NEXT H
120 END
```

Making graphics displays is an interesting topic for many students of all ages. Don't reserve the use of graphics as though it were a difficult topic. As with so many aspects of programming, just give students the fundamentals and get out of the way.

Next month we'll examine a graphics display that ties together all those ideas discussed in this one. Would you believe a hovering space ship and user-controlled ground-to-air missiles? Don't miss it.□

Walter Koetke, Putnam/North Westchester BOCES, Yorktown Heights, NY 10598.

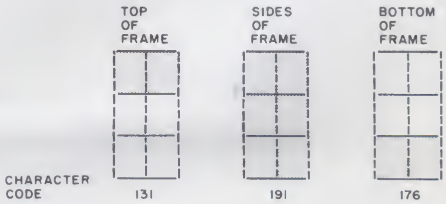


Fig. 1.

By Eric Maloney

Revolutionary Times

Overthrow Of Language Plotted?

Americans have always been partial to revolutions: industrial, political, social and otherwise. But lately, things have been getting out of hand. We're in the throes of so many revolutions, it's surprising we have time for anything else.

For example, a recent article in *Business Week* describes the *home information revolution*. This is similar to the coming *revolution in home information retrieval*, analyzed in the book *Videotext*. These revolutions are tied in to the *home computer revolution*, described on the jacket of Robert L. Perry's book, *Owning Your Home Computer*.

Perry himself explores the *Information Revolution*, which he says is closely related to the *calculator revolution*.

U.S. News and World Report says that America's newspapers are being confronted by a *technological revolution*, while N. B. Nannay, a vice-president of research and patents at Bell Labs, asserts that changes in our society are being caused by the *electronics revolution*.

Business Week, not about to box itself into a corner, refers in another issue to the *microelectronics revolution*, which "already is transforming virtually every facet of life at breathtaking speed."

But under the rules of American free enterprise, the revolution to meet with the most success will probably be the one that is marketed properly. That, according to Herman Kahn of the Hudson Institute, is the *computer revolution*, which he says is "the most advertised revolution in world history." It's hard to beat a good publicity campaign.

* * * * *

Speaking of overworked phrases and words, why does every subroutine have to be handy and little? Why are all new compilers and word processors powerful? Why are all disk operating systems flexible? For once I'd like to see a press release announce a software product that was weak and rigid.

* * * * *

The Intergovernmental Bureau of Informatics is putting together a $1 billion program "for the information of the Third World." Why? Because, says IBI, "the informatization of underdeveloped countries is a necessary condition . . . for their development."

I pity the poor fellow who has to translate that into Kenyan.

* * * * *

Jim Howell of San Jose, CA, excoriates me for disapproving of the use of keyboard as a verb while using lowercase, as in "Microcomputing lowercases ac . . . ." Actually, one of our assistant editors

R/G TECHNO-MULTI-MICRO-GRAPHIC-COM-FLEX-CALC-NET 2000
#3 Disko Duck Drive
Silicon Valley, California

June 25, 1981

Erik Maloney
c/o MICROCOMPUTING magazine
80 Pine Street
Peterborough, NH 03458

Sir:

We at RGTMMGCFCN2000 take complete umbrage with your disclosure of our proprietary process for establishing trade names. Since the process is, or has been, unfathomable by the public, heretofore, we must assume you were given information by person or persons who held this information in confidence.

Therefore, we believe a prima facie case of industrial espionage has been generated (note passive voice), and are turning the matter over to our solicitors, International Legal Research Systems Group, Ltd., who, if they have the time will be in touch. Otherwise a very ugly little man will peek through your window.

Parity errors on you and yours, sir!

Hy Teck,
Director of Addle-Brained Research.

raised a similar objection when proofing the article in galley form. My rationale is, and remains, that while lowercase may be awkward as a verb, it's all we've got—keyboard can be replaced by the much more acceptable word type.

Howell also mentions that in the process of criticizing the use of the term infant mortality when describing early failure of electrical parts, I implied that Percom and Infoscribe—whose press releases I quoted—were responsible for its invention. I had no intention of indicating such —as Howell points out, the term "has been around as long as integrated circuits have, perhaps longer."

But then, so has the bubonic plague.

* * * * *

The mail has been getting very strange lately. Take, for example, the one shown here. Is it too much sun? Something in the water? I'm glad my mother warned me not to go to California . . .

* * * * *

The long-awaited (by us, anyway) "How to Write for Kilobaud Microcomputing: A Guide for Beginners and Pros" is hot off the presses. We've expanded it to four pages, and it's chock-full of hints on how to prepare and write an article for this magazine. For a copy, send a self-addressed, stamped envelope to Writer's Guidelines, Microcomputing magazine, Pine St., Peterborough, NH 03458.□

COMPUTER CLINIC

I am looking for any information on the Commodore VIC-20 computer. It is officially on the market, but no one seems to know anything about it.

Aaron Contorer
1521 Central Ave.
Deerfield, IL 60015

I wanted to buy SWTP's cassette 4K and 8K BASIC, but they told me it's no longer available. I am willing to pay a fair price for a copy of each with documentation (I have to change it to run with MIKBUG II).

C Pilipauskas
6426 S. Fairfield Ave.
Chicago, IL 60629

I am looking for a FORTRAN compiler, to run with the PolyMorphic System 8813.

Joachim Zeller
c/o Dr. Kurt Schumacher, Ring 4
6120 Michelstadt 5
West Germany

I am working on a problem concerning home computers with the National Futuristics Bowl. I would greatly appreciate any information about this subject, and any problems anyone knows of that deal with home computers.

Paul Louie
c/o Tuckahoe TAG Center
9000 Threechopt Road
Richmond, VA 23229

MICRO QUIZ

What Does This Program Do?

After the following program is executed, what is the value of variable z?

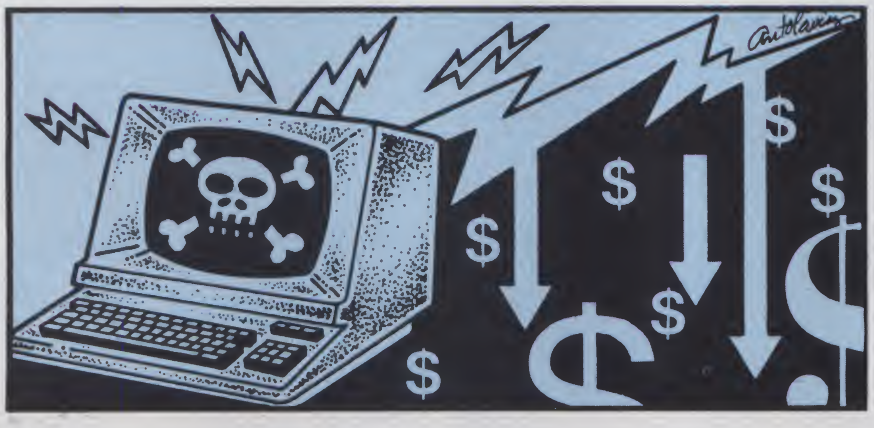
```
z = 25
for j = 0 to 25 step 1
    if (j/3 = int(j/3)) then z = z - j
    if (j/4≠int(j/4)) then z = z - 2
next j
```

(answer on page 212)

Compiled by Eric Maloney

Malprogram Suits

No Rock To Hide Under



Running for Coverage

You're a computer consultant and have designed a software program for hospital or clinical testing. Its diagnosis of a laboratory sample is incorrect—resulting in patient inconvenience or serious illness.

Or perhaps your software designs a mathematical model of structures for a small construction firm. Its errors cause a building to collapse, with loss of material and maybe loss of life.

It's bad enough when your software package doesn't perform for a customer. But what if its errors caused physical injury or property damage?

Computer law experts are seriously considering the effects of software error. They say programmers have little protection from liability suits, because no insurance is available and courts have not been upholding product disclaimers.

What programmers need is an errors-and-omissions policy—malpractice insurance for professionals not having direct contact with the body as physicians do. It covers negligence and strict liability, which does not involve intended harm.

Data processing insurance for businesses who depend on computer systems has been standard for over a decade. Coverage includes business interruption, which reimburses company losses, and extra expense, which covers the time and labor to do computer tasks manually after a breakdown. DP policies usually insure company software from electromagnetic erasure up to replacement cost.

But no companies are offering errors-and-omissions coverage for software programmers, though some are investigating the market. Those interested include St. Paul Insurance and the brokerage firm Frank B. Hall, Inc.

"It's a brand new market," says Thomas Cornwell of Chubb and Son, Inc. "There are no statistics. We don't know whether we'll lose our shirts or not."

Roy N. Freed, a computer law expert, says insurers are reluctant to establish a policy until they know what their losses are going to be. Instead of waiting, they should be astute enough to make that judgment now, he says. "You don't have to look at experience. You can project, extrapolate." Companies should be comparing the consequences of computer operations with those of manual tasks.

But insurance companies just don't know what claims to expect. Freed has thought of possible disasters ranging from factory explosions to harmed patients. These make insurers nervous.

"When you start getting into scientific areas, it becomes extremely difficult to find companies willing (to insure)," says Guy Migliaccio of the brokerage firm Marsh & McLennan.

Some companies will insure the businesses and hospitals themselves against system or program failure, but they won't insure the programmer against liability suits.

Freed thinks more claims will be filed against programmers because hospitals have charitable and governmental immunity, protecting them against liability suits. The manufacturer or designer of the program is the next logical defendant. Programmers could also be sued by doctors for injury to reputation.

Computers are used widely in hospitals, to monitor intensive care patients, process laboratory tests and diagnose illness. Question-and-answer programs determine a patient's disease by the symptoms given to the computer; a Salt Lake City hospital's computer can pinpoint up to 140 different ailments. Patient monitoring systems track brain waves, blood pressure and heart waves.

Physical simulation systems react to medical procedures as a patient would.

Other computer programs give mathematical models of chemical reactions. Both are used primarily in medical schools and research centers.

Engineers are also using programs that "enable people to design and structure more precisely and save a lot of money," says Freed. These programs produce mathematical models of structures to aid engineering calculations.

Though no suits have been filed against programmers for bodily injury or property damage, some have been held liable for human error in the use of software. In New Zealand a navigational program was fed the wrong information. Though it was only two degrees and ten minutes of longitude off, the plane crashed, killing all 257 aboard. Air New Zealand employees were investigated by the Royal Commission of Enquiry and several may face criminal charges.

The design of an engineering program also was blamed partly for the roof collapse of the Hartford, CT, Civic Center. The engineering firm that designed the program is being sued in the wake of $14 million in property damages.

The Jury is Out

The liability of software programmers remains untested in the courts, however. And judges have been inconsistent in upholding product disclaimers. "In today's computer world, contracts are accepted until you get in front of a judge," says Ed Saltzberg, a lawyer who practices computer law. One Maryland appeals court threw out the same disclaimer that was upheld in an Arizona appeals case.

Service bureaus—companies that provide data processing services for others—are more protected than the lone programmer, because service bureaus write specific disclaimers. "The individual out there who wants to program is more vulnerable than full-sized service bureaus," says Bob Ramstad of St. Paul Insurance. Some courts uphold disclaimers as binding contracts between two parties, and the service bureaus usually have the most detailed provisions.

Malpractice and errors-and-omissions cases usually fall outside disclaimers anyway, says Thomas Christo, a lawyer who lectures on the liability of data processing vendors. Courts have also struck down disclaimers in cases involving innocent misrepresentation (when a product is said to perform a function it cannot, whether the company knows it or not) and fraud.

"A disclaimer is never enough, and it's not going to be enough," says Ed Saltzberg. Often errors-and-omissions cases involve third parties to whom contracts don't apply, such as hospital patients.

If big businesses were aware of the risks, they would be demanding errors-and-omissions coverage, says Freed. But the only software lawsuits have involved breach of contract or failure to perform. Because property damage and bodily injury have not yet been tested in the courts, programmers don't see the need for insurance.

Bob Ramstad says he's "not getting what I would call really positive vibes" from the dozen programmers he's talked to. Most tend to think the cost would outweigh the benefits, he says.

Insurance companies are having a hard time designing policies because software is hard to define.

An errors-and-omissions policy probably would carry a hefty deductible. Fred Nagel of the San Francisco Insurance Center estimates a deductible of $5000 to $10,000; it could run into six or seven figures, says David Z. Webster of Frank B. Hall, Inc. The premium might be "anywhere from one half of 1 percent of indemnity to 10 percent of indemnity," Webster says.

Insurance companies are having a hard time designing policies because software is hard to define. "Lack of understanding is endemic," says Freed. "Insurance people think in terms of categories," he says, and the categories aren't appropriate. Before computers, negligence and liability referred to acts of people, not machines.

It would be easier for insurers to design blanket policies with exceptions spelled out. This would protect the user and programmer more than a traditional policy with inappropriate language.

Other Ways to Protect

But insurance is not the only way to deal with liability, says Freed, who recommends a risk management program including quality control. Companies should recall defective items and provide backup for their customers, he says. Users should check their programs carefully to avoid harm.

Programmers can also protect themselves by having customers waive claims. Such waivers could limit monetary claims, exclude monetary damages altogether or specify the type of liability waived—negligence or products liability, for example. Freed also advises waiving the implied warranties of the Uniform Commercial Code, which state that products must be suitable for ordinary use and meet label specifications.

Another type of waiver is "exclusion of remedy," which prohibits the client from seeking legal action. Exclusions can be specific or apply to everything but repair or replacement.

Programmers should also work toward legislation that would limit liability claims, Freed says, citing similar laws protecting banks and automotive companies.

Insurance is still an option, but with only a handful of companies interested, its development appears far off. Programmers could go to a specialty house, such as Lloyd's of London, to seek customized policies.

In the meantime, programmers should become aware of possible liability and find other ways to cover themselves. Disclaimers do not protect against property damage and bodily injury, so waivers of liability and quality control are a programmer's best bet.

As programmers become aware of the risks they incur, they may pressure insurance companies to develop errors-and-omissions policies. But the cost of premiums and the high deductible could be prohibitive to the small programmer.

Contributed by Betty Thayer,
***Microcomputing* Staff**

Exploring the MC6809

The Boston Globe's ad described the seminar as an "MC6809 Microprocessor Course by Motorola." It sounded like a chance to find out what a high technology training session is like—a chance to rub elbows with the engineers who are designing equipment on the leading edge of the computer industry.

And since Radio Shack has elected to use this chip (actually a later model, the 6809E) in its new color computer, it also seemed like a good opportunity to find out what made that 6809E tick.

The course was scheduled for June 2–5 in Burlington, MA, just off Rte. 128 outside Boston. I was intimidated because I don't care for heavy duty commuting around big cities. But I gritted my teeth, gave my break pads a workout on the way down from New Hampshire and made it to the Burlington Holiday Inn only 15 minutes late.

The architect who designed Holiday Inns 25 years ago was apparently in a perpetual state of depression. The place is gloomy and dark and tries to overcome these deficiencies with patterned carpets and draperies that are positively garish. Nevertheless, I took my place alongside the 16 others who had plunked down $430 to learn about what Motorola calls "Innovative Systems Through Silicon."

Upon arrival, the course instructor, Ray Doskocil of Motorola, handed me a

TWELVE STRONG HEATH/ZENITH YOUR

Pick a strong partner

A computer purchase is the beginning of a long term partnership between you and the people you buy from. Your ongoing need for software and accessories requires a partner who will stand by you with a growing line of products. And nowhere will you find a more complete line of hardware, software and accessories than at your Heathkit Electronic Center. Here are twelve strong reasons to make Heath/Zenith your partner.

1. The All-In-One Computer

The heart of the Heath/Zenith line is the stand-alone *89 Computer*. It's a complete system with built-in 5¼-inch floppy disk drive, professional keyboard and keypad, smart video terminal, two Z80 microprocessors, and two RS-232C serial I/O ports. It comes with 16K RAM, expandable to 64K.

2. Peripherals

These include the popular *Heath/Zenith 19 Smart Video Terminal*, loaded with professional features. And the *14 Line Printer*, priced as low as $495. Other printer brands are on display, including high-speed, typewriter-quality printers.

3. Software

Word processing, includes reliable, easy-to-use Zenith Electronic Typing and powerful, full-featured WORDSTAR.

Small Business Programs, feature General Ledger and Inventory Control.

HUG, Heath Users' Group, offers members a library of over 500 low-cost programs for home, work or play.

4. Programming Languages

For your own custom programs, Microsoft languages are available in BASIC (compiler and interpreter), FORTRAN and COBOL.

5. Operating Systems

Three versatile systems give you the capability to perform your specific tasks.

CP/M by Digital Research makes your system compatible with thousands of popular CP/M programs.

UCSD P-System with Pascal is a complete program development and execution environment.

HDOS, Heath Disk Operating System gives you a sophisticated, flexible environment for program construction, storage and editing.

6. Utility Software

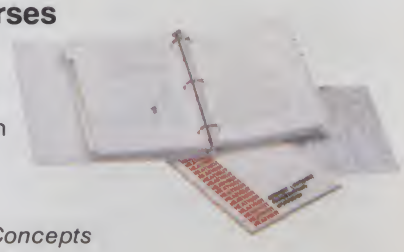
Expand the performance range of your computer with a broad selection of utility tools, including the best of *Digital Research* and the complete line of innovative *Softstuff* products.

7. Disk Systems

The 8-inch Heath/Zenith 47 Dual Disk System adds over 2 megabytes of storage to your 89 Computer. Diskettes are standard IBM 3740 format, double-sided, double-density.

The 5¼-inch 87 Dual Disk System adds 200K bytes of storage to your 89. Both disk systems feature read/write protection and easy plug-in adaptability.

8. Self-Study Courses



Learn at your own pace with *Programming Courses* that teach you to write and run your own programs in Assembly, BASIC, Pascal or COBOL.

A course on *Computer Concepts for Small Business* gives you the understanding to evaluate the ways a computer can benefit your business.

Personal Computing is a complete introduction to the fundamentals for the novice. Every Heathkit/Zenith course is professionally designed for easy, step-by-step learning.

All Heath/Zenith Computer Products are available completely assembled and tested for commercial use. Or in easy-to-build, money-saving kits.

✓236

REASONS TO MAKE COMPUTER PARTNER

9. Expansion Options

Communicate with the outside world through a *Three-port EIA RS-232C Serial Interface*.

Expand RAM to 64K with easy-to-install *expansion chips*.

10. Accessories

Your Heathkit Electronic Center has the latest in modems, black-and-white and color video monitors, computer furniture and a full line of supplies, accessories, books and parts.

11. Service

No one stands by you like Heath/Zenith. We help you get your system up and running smoothly. Service is available from trained technicians, over the phone or at one of 56 Heathkit Electronic Centers.

12. Value

Your money buys you more because Heath/Zenith prices are among the industry's most competitive. Make your own comparison and find out how much you can save.

Complete, integrated computer hardware and software, designed to serve you and to grow with you – that's what to look for in a strong partner. And with Heath/Zenith you get it all under one roof.

All at your Heathkit Electronic Center

Pick the store nearest you from the list at right. And stop in today for a demonstration of the Heath/Zenith 89 Computer System. If you can't get to a store, send $1.00 for the latest Heathkit® Catalog and the new Zenith Data Systems Catalog of assembled commercial computers. Write to Heath Co., Dept. 351-814 Benton Harbor, MI 49022.

Visit Your Heathkit Electronic Center*

where Heath/Zenith Products are displayed, sold and serviced.

PHOENIX, AZ
2727 W. Indian School Rd.
602-279-6247

ANAHEIM, CA
330 E. Ball Rd.
714-776-9420

CAMPBELL, CA
2350 S. Bascom Ave.
408-377-8920

EL CERRITO, CA
6000 Potrero Ave.
415-236-8870

LA MESA, CA
8363 Center Dr.
714-461-0110

LOS ANGELES, CA
2309 S. Flower St.
213-749-0261

POMONA, CA
1555 N. Orange Grove Ave.
714-623-3543

REDWOOD CITY, CA
2001 Middlefield Rd.
415-365-8155

SACRAMENTO, CA
1860 Fulton Ave.
916-486-1575

WOODLAND HILLS, CA
22504 Ventura Blvd.
213-883-0531

DENVER, CO
5940 W. 38th Ave.
303-422-3408

AVON, CT
395 W. Main St. (Rt. 44)
203-678-0323

HIALEAH, FL
4705 W. 16th Ave.
305-823-2280

PLANTATION, FL
7173 W. Broward Blvd.
305-791-7300

TAMPA, FL
4019 W. Hillsborough Ave.
813-886-2541

ATLANTA, GA
5285 Roswell Rd.
404-252-4341

CHICAGO, IL
3462-66 W. Devon Ave.
312-583-3920

DOWNERS GROVE, IL
224 Ogden Ave.
312-852-1304

INDIANAPOLIS, IN
2112 E. 62nd St.
317-257-4321

MISSION, KS
5960 Lamar Ave.
913-362-4486

LOUISVILLE, KY
12401 Shelbyville Rd.
502-245-7811

KENNER, LA
1900 Veterans Memorial Hwy.
504-467-6321

BALTIMORE, MD
1713 E. Joppa Rd.
301-661-4446

ROCKVILLE, MD
5542 Nicholson Lane
301-881-5420

PEABODY, MA
242 Andover St.
617-531-9330

WELLESLEY, MA
165 Worcester Ave.
617-237-1510

DETROIT, MI
18645 W. Eight Mile Rd.
313-535-6480

E. DETROIT, MI
18149 E. Eight Mile Rd.
313-772-0416

HOPKINS, MN
101 Shady Oak Rd.
612-938-6371

ST. PAUL, MN
1645 White Bear Ave.
612-778-1211

BRIDGETON, MO
3794 McKelvey Rd.
314-291-1850

OMAHA, NE
9207 Maple St.
402-391-2071

ASBURY PARK, NJ
1013 State Hwy. 35
201-775-1231

FAIR LAWN, NJ
35-07 Broadway (Rt. 4)
201-791-6935

AMHERST, NY
3476 Sheridan Dr.
716-835-3090

JERICHO, L.I. NY
15 Jericho Turnpike
516-334-8181

ROCHESTER, NY
937 Jefferson Rd.
716-424-2560

N. WHITE PLAINS, NY
7 Reservoir Rd.
914-761-7690

CLEVELAND, OH
28100 Chagrin Blvd.
216-292-7553

COLUMBUS, OH
2500 Morse Rd.
614-475-7200

TOLEDO, OH
48 S. Byrne Rd.
419-537-1887

WOODLAWN, OH
10133 Springfield Pike
513-771-8850

OKLAHOMA CITY, OK
2727 Northwest Expressway
405-848-7593

FRAZER, PA
630 Lancaster Pike (Rt. 30)
215-647-5555

PHILADELPHIA, PA
6318 Roosevelt Blvd.
215-288-0180

PITTSBURGH, PA
3482 Wm. Penn Hwy.
412-824-3564

WARWICK, RI
558 Greenwich Ave.
401-738-5150

DALLAS, TX
2715 Ross Ave.
214-826-4053

HOUSTON, TX
1704 W. Loop N.
713-869-5263

SAN ANTONIO, TX
7111 Blanco Road
512-341-8876

MIDVALE, UT
58 East 7200 South
801-566-4626

ALEXANDRIA, VA
6201 Richmond Hwy.
703-765-5515

VIRGINIA BEACH, VA
1055 Independence Blvd.
804-460-0997

SEATTLE, WA
505 8th Ave. N.
206-682-2172

TUKWILA, WA
15439 53rd Ave. S.
206-246-5358

MILWAUKEE, WI
5215 W. Fond du Lac
414-873-8250

*Units of Veritechnology Electronics Corporation in the U.S.

Prices and specifications subject to change without notice.

HEATH/ZENITH ✓236

Your strong partner

✓See List of Advertisers on page 210

small library of books, data sheets and other materials. The class consisted of men from companies around the Boston area, companies like Polaroid, IBM and Simmonds Precision. There were also a couple of out-of-town class members from the Oldsmobile Division of General Motors. GM is using microprocessors to monitor fuel efficiency, interior comfort levels and other factors in some of their new top of the line models. I began to get the feeling that this was a course for heavyweights in microelectronics.

The course started out, however, with a review of number systems and how to convert from decimal to hexadecimal to binary, how to add and subtract binary numbers, etc. Surely, I thought, a course that began with such an elementary level wouldn't get too involved—but that turned out to be Falacious Assumption No. 1.

From there, we plunged right into a description of some of the chips sold by Motorola—the 24- and 18-pin static RAM and 16-pin dynamic RAM chips, the mask programmable ROMs and ultraviolet erasable PROMs, the 6800 family of peripheral chips—the timer chips, modulators, video display generators, etc.—and finally the eight-bit 6809 microprocessor unit (MPU) itself. The architecture of the chip was described in detail, including the function of each of the chip's 40 pins (see Fig. 1), its internal registers (there are four eight-bit registers and five 16-bit registers) and the timing device (a crystal oscillator is tied to the system to act as the clock). Next, there was a description of the 6809 universe (a 65536 byte addressable memory) and a brief overview of the instruction set for the device.

As lunchtime approached that first day, I found that I was totally absorbed in an area I had no previous interest in and felt was beyond my powers of comprehension. Either Ray Doskocil was doing something right or I was discovering myself to be a closet technocrat!

The major strength of the MC6809E microprocessor turns out to be its addressing modes. In addition to the common addressing modes (direct, immediate, inherent, relative, extended and indexed), the 6809E allows two indirect addressing modes. The extended indirect and the indexed indirect addressing modes make the chip extremely versatile. Doskocil pointed out, however, that versatile usually equates with complicated, thereby heading off Falacious Assumption No. 2.

The large choice of addressing modes is made necessary because the 6809E uses memory-mapped I/O. This means that the microprocessor is connected to its peripheral devices using memory-mapped interfaces. These interfaces are addressed at specific predetermined locations in the MPU's memory universe. Therefore, anytime the 6809E communicates with the outside world, it merely thinks it is reading data in its memory. This memory-mapped I/O increases the speed and efficiency of the processor but necessitates some contortions in addressing.

For example, Radio Shack's new Color Computer uses a video display generator to print characters on a video screen and two peripheral interface adapters for keyboard and joystick I/O. Each PIA contains six registers—two registers each for data, data direction and control. When the software calls for data from a PIA, the control registers select which data register will load data from the keyboard or joystick. The PIA is tied to the MPU via the memory address lines plus the data lines, and the MPU can access the data in the PIA data registers in the same way it would load data from a RAM location. The video display generator has similar data registers to conduct affairs between the MPU and the video circuitry. This is the essence of what Motorola's instructors call memory-mapped I/O.

Addressing modes for the 6809E turned out to be such a mind-boggling topic that by 4:11 PM on Day One, my interest had slipped to the point that I had to work hard to fight off fits of hysterical giggling. I attributed this to a slight overdose of bits.

Day Two

Day Two found traffic no better, but since I was now a seasoned commuter, I arrived on time.

The entire morning session was spent reviewing the material covered on Day One. The only problem was that some of the material reviewed was *not* covered on Day One. I made a note not to miss a review.

The rest of the day was devoted to a description of the Motorola PIA (peripheral interface adapter) and the ACIA (asynchronous communications interface adapter). Students were instructed in the techniques of designing a hardware system, wiring up the parts and providing the software control to select components (using chip select pins on each part). This part of the course was especially well-developed and presented.

Hardware design is obviously not a subject that can be learned in one training session. But this Motorola workshop included an excellent introduction to system configuration. For the microprocessor to select the appropriate RAM, PIA or other device, there must be a unique address for that device. Using the MPU address bus, peripheral devices can be located anywhere in the 6809E memory universe and addressed directly by software. This is done using the chip select pins on each of these devices.

The chip select pins can be set either high or low and several of these pins together, tied to middle- and upper-address lines, can produce a unique address. This process permanently defines the addresses for all RAM, ROM and other peripheral devices (i.e., RAM at 0000–03FF, ROM at FE00–FFFE, etc.). The description of this wiring-up a chip select process was a highlight of the course.

After a review of the previous day's

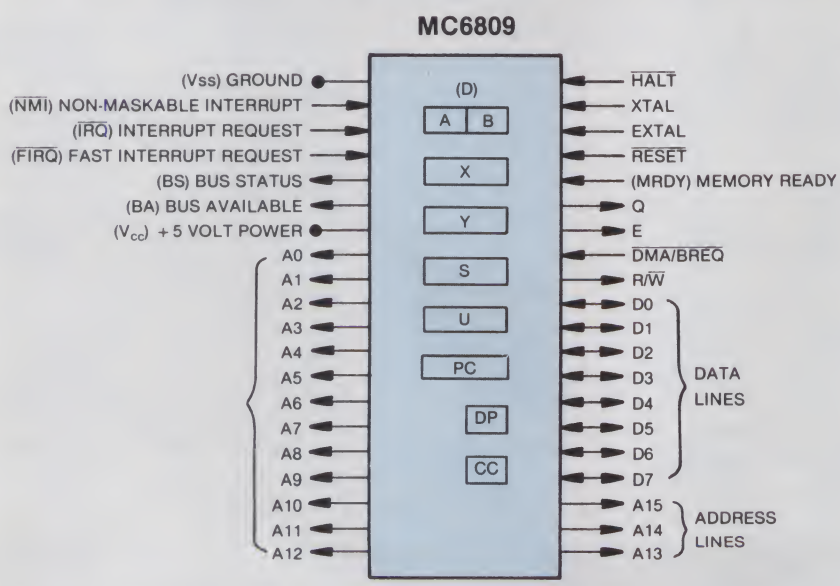


Fig. 1. The special function pins for the 6809 MPU include (1) the XTAL and EXTAL pins which connect to the external clock. These inputs produce the system pulses which are output on the Q and E pins. (2) The DMA/BREQ pin is a direct memory access/data bus request pin used for I/O that must be accomplished at pulse rates slower than that for which the system is configured. (3) The NMI, IRQ and FIRQ pins are interrupt pins that allow interruption of the MPU for multiple task operation. Interrupts cause internal registers to be dumped to the stack so that the processor can perform the interrupt service routine and then return to pick up its original task where it left off. (4) The bus status and bus available pins are the 6809's traffic cops, keeping the data and address buses free of traffic jams.

work, Day Three tackled the 6809 instruction set and the Motorola macro assembler designed for the development system called the Exorcisor. (The Exorcisor is a microcomputer with video terminal, keyboard, floppy disk drive and printer, used primarily to develop 6800 family software.) Rather than being a primer in machine-language programming, however, this part of the course concentrated on software control of hardware processes. A variety of sample programs was dissected and discussed to explain the use of the individual instructions in the instruction set. We were given a sales pitch about the 6800 family of devices, including memory management units which allow expansion of the 6809E memory space from 64K to 2 megabytes, floating points ROMs and the like.

Day Four was a lab session in which students were encouraged to write a simple program and type it into an Exorcisor using the system's Monitor program. Then the program was assembled and run. Fortunately, your program didn't have to run for you to graduate.

Conclusions

The course required a background in digital electronics and some knowledge of logic elements and programming concepts. For hardware and software engineers, the course was a solid introduction to the use of Motorola's products. For the beginning machine-language programmer or the novice hardware basher, the course and its price tag may be a bit lofty.

At the conclusion of four intensive days of training, a couple of students were overheard to comment that they had decided that, for the moment, the best way to address the MC6809E was the "irrelevant" mode. I'll let you decide what this tells you about an engineer's sense of humor.

Contributed by G. Michael Vose, *Microcomputing* Staff

Course Schedule.

Sept. 29,30	River Edge, NJ
Oct. 1,2	Rochester, NY
Oct. 20,21	Houston, TX
Oct. 22,23	Dallas, TX
Nov. 10,11	San Jose, CA
Nov. 12,13	Denver, CO

To enroll in any course, call Motorola Technical Training, 602-962-2345 or 602-244-4945.

VDT Bill Defeated

A first-in-the-nation bill to impose video display terminal health and safety regulations on employers has been defeated in the Maine legislature, but its sponsor promises that the bill will reemerge before the new legislature in 1983.

"If I'm still here it will be introduced. If I'm not here it'll be introduced. I'm a persistent broad and my fellow legislators know that," says Edith Beaulieu, a Democrat from Portland.

The bill, which would have concerned employees who work on VDTs at least four hours a day, would have required employers to provide annual eye exams, semiannual maintenance of equipment, a rest period or a change of tasks every two hours and literature on proper VDT use.

Though the labor committee, which Beaulieu chairs, recommended passage of the bill by an 8–5 count, the bill was defeated by 2–1. The arguments were that it was not timely and needed more study.

The bill. . . would have required employers to provide annual eye exams. . . (and) a rest period or change of tasks every two hours.

"Most of the arguments claimed that the studies were not acceptable, that there was no definite relationship between VDTs and eye problems," Beaulieu says.

But she says she isn't disappointed by the results.

"It was a first effort anywhere in the country," she says. "I think we did a hell of a job educating both employers and employees."

Beaulieu became interested in the problem as a shop steward from the Newspaper Guild. She is a cleaning woman for Gannett's Portland papers—the *Press Herald*, *Evening Express* and *Sunday Telegram*—and was concerned about the introduction of VDTs in the newsroom. She also read about problems at other newspapers in the Guild's newsletter.

VDT health and safety has increasingly become an issue with unions. Both unions and scientists say that poorly designed equipment and workplaces can cause visual impairment, stress, musculoskeletal problems, anxiety and fatigue.

For a complete discussion of the VDT issue, see "Video Dismay Terminals" on page 43 of the July *Microcomputing*.

New Publications

Several microcomputer publications have been released recently. A few that have come our way this month include:

- Omni Software Systems, Inc., has released the North Star Business Software catalog. It lists over 20 business and utility programs that run under North Star DOS and BASIC. Documentation is available separately. The catalog is free by writing or calling Omni Software Systems, Inc., 146 North Broad St., Griffith, IN 46319 (219-924-3522).
- In January 1981 the College of Education at Arizona State University hosted a conference designed to introduce educators and administrators to the classroom applications of microcomputers. The conference proceedings are now available in a 340-page book that includes over 30 articles. "Instructional Techniques for Teaching BASIC Programming to Elementary Children," "Using Computers with Blind and Deaf Children" and "Microcomputers in High School Physics" are a few of the titles included. Make your check for $10 payable to Arizona State University and send it to Dr. Gary Bitter, Arizona State University, Payne B203, Tempe, AZ 85287.
- *Chromasette Magazine* is a monthly magazine for the TRS-80 Extended BASIC Color Computer on cassette tape. The cassette contains six to eight programs including games, tutorials and utilities. Included with the cassette, on paper, is information on the programs. *Chromasette* costs $45 a year. For more information call or write *Chromasette Magazine*, PO Box 1087, Santa Barbara, CA 93102 (805-963-1066).
- *Computer Sales Digest* is a monthly 12-page publication in a newsletter format designed to help computer companies market their products more successfully. The newsletter is published by Datasearch, Inc., a source of computer industry sales and marketing publications, services and seminars since 1977. Feature sections of the letter will cover selling techniques, promotion, marketing and business opportunities. Regular subscription rates are $95 a year; there is a special $60 rate for prepaid charter subscribers. Datasearch, Inc., is located at 4954 William Arnold, Memphis, TN 38117 (901-761-9090).
- *The Community Computerist's Directory*, "the who's who of people and computers," is the white and yellow pages for the computer world. It's a national paper database in a phone book format for all computer users, including beginners, professionals, businesses and prospective computer buyers.

The CCD is published twice a year, in January and June, by Alternet, Inc. Quarterly updates will begin in April 1982. The current issue, July–December 1981, is $3.50; future issues will be $5. Subscriptions are $10 a year and include a free white page listing, two issues and all quarterly updates.

The white pages contain over 230 noncommercial listings from individuals and organizations interested in sharing information, skills and resources. The yellow pages have over 580 entries with listings and ads under 72 subject headings including hardware, software, databases, consultants, systems houses, publications and services. In addition, there is a glossary of computer terms; a bulletin board section with 380 private CBBSs, ABBSs and Forum-80s; and a user group section that lists 250 amateur and professional groups. All listings can be found by geographic location, alphabetically and by key words. For more information write to The Community Computerist's Directory, PO Box 405, Forestville, CA 95436 (707-887-1857).

Communications and the Future

"Communications and the Future" will be the topic of the Fourth General Assembly of the World Future Society, July 18–22, 1982, at the Sheraton Washington Hotel in Washington, DC.

Subjects to be covered will include television, computers, electronic data systems, networking, body language, satellite broadcasting, books, motion pictures and ordinary conversation.

A few of the specific topics being considered for the assembly are: Home Computer Terminals: Instant Information About Everything; Privacy in Modern Information Systems; Information Overload: Coping Strategies for a Growing Problem; Loss of Cultures in the Information Age; Bringing Nutrition Information to the Third World; The Computerized Farm; Satellites and Communications; Centralization vs Decentralization: The Role of Communications; and Reviving Rural Communities Through Telecommunications.

Members of the Society are encouraged to submit papers on communications for the Assembly volume, a publication that will be distributed at the conference free of charge to all those who register for the entire five days.

The conference format will give participants a wide range of activities, including concurrent general sessions and opportunities to meet with speakers in informal group sessions. In addition, exhibits at the conference are expected to display the latest in communication technology.

Plans call for the Fourth General Assembly to be electronically linked to the 1982 Society for International Development conference in Baltimore, MD, which will be held at the same time as the General Assembly.

The upcoming conference is expected to be the largest in the Society's history and early registration is encouraged. There are significant savings and additional benefits for those who register early. Also, if your plans change, you can receive a full refund up to April 30, 1982.

Comments, ideas, suggestions and proposals for speakers and topics should be addressed to the Assembly Coordinating Committee, World Future Society, 4916 St. Elmo Ave., Washington, DC 20014. Cost for nonmembers is $170 by Dec. 31, $185 by March 31, $200 by June 30 and $215 after June 30. Fees should be sent to General Assembly Registration, in care of the above address.

Software for Logo

The Massachusetts Institute of Technology is accepting applications from software distributors for the licensing of the Apple Logo Educational Software System.

A review of the prototype of the system is available in the June, 1981 issue of *Byte* magazine.

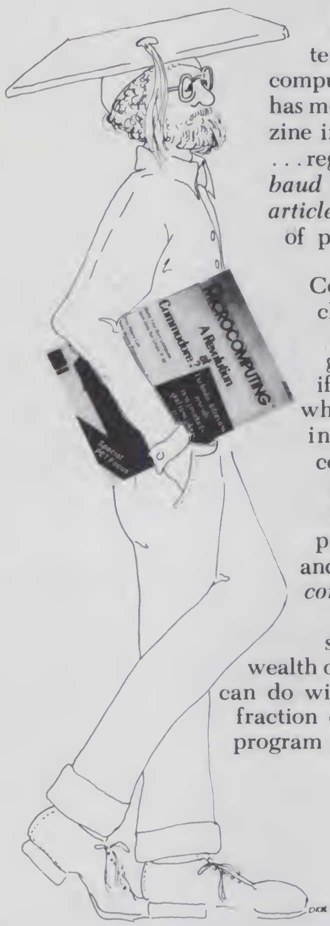
Interested companies should contact the Patent, Copyright and Licensing Office, MIT, 77 Massachusetts Ave., Room E19–722, Cambridge, MA 02139.□

✓See List of Advertisers on page 210

Are you looking for . . .

MEAT or FAT?

When you subscribe to a magazine, you want to get REAL SOLID INFORMATION, not just a giant catalog of ads every month . . . and mostly the same ads, if you've noticed. *Kilobaud Microcomputing has the meat:* feature articles written by the most knowledgeable people in the field, yet written for the relative newcomer to computing. *Kilobaud Microcomputing* has more articles than any other magazine in the field . . . by a wide margin . . . regardless of fatness. In 1980 *Kilobaud Microcomputing* published *409 articles* . . . and that included a wealth of programs which you could use.

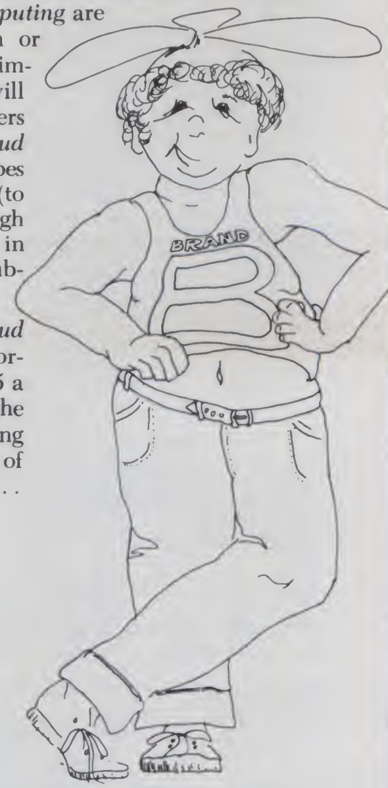


Compare that with 133 paltry articles in *Brand B*, the "*Fat Albert*" of the computer field. You can get far more from your computer if you can really understand it, which is where the simple articles in *Kilobaud Microcomputing* come in. You don't need a science degree to get through it like some magazines I could mention. The practical reviews of both hardware and software in *Kilobaud Microcomputing* can save you a bundle . . . far more than the cost of a subscription . . . even for life. The wealth of programs give you things you can do with your computer . . . again at a fraction of the cost of buying the same program over the counter. Further, the articles on programs *help you learn* how to write and modify programs that you have to do yourself.

When you subscribe to a magazine, you want to pay for the information, not a bunch of ads. The advertisers are already paying for them so why should you? *Kilobaud Microcomputing* has been running around 40% advertising while *Brand B* has been running 60-70%, making fat issues, but with *little real information* for you.

You want to learn about computers as fast as you can. The editors of *Kilobaud Microcomputing* are under orders (pain of death or worse) to keep the material as simple as possible so new comers will be able to learn about computers as quickly as possible. *Kilobaud Microcomputing* covers all types of microcomputers, including (to some extent) the TRS-80 though this is covered overwhelmingly in *80 Microcomputing*, a sister publication.

At $2.95 a copy, *Kilobaud Microcomputing* is the best information buy you'll find. At $25 a year (you save $10.40 off the newsstand price) you're investing in the most valuable library of microcomputing you can buy . . . 2,960 pages in 1980!



I WANT MEAT!

☐ **So, please bill me $25 for one year's subscription to *Kilobaud Microcomputing.***

Canadian $27/1 year only, US funds.
Foreign $35/1 year only, US funds.
Please allow 4-6 weeks for delivery.

name ____________________

street ____________________

city ____________ **state** ________ **zip** ______

Kilobaud Microcomputing ● Box 997 ● Farmingdale NY 11737

is a division of GREEN **Peterborough NH 03458**

319B7

LETTERS TO THE EDITOR

Bloody Awful Software

I have just written to Texas Instruments indicating my displeasure over the bloody awful software material they've supplied with their TM 990/189 single board computer. I have been sucking my teeth for some time now trying to fathom the convoluted explanations and "examples" given in the TI manual for this latest miscarriage of "high" technology.

For example: A simple program like

```
10 Let J = 12345
20 Print J
30 Stop
```

does not print J on the display of the 990/189. It is implied that if one had connected an external terminal to the board, it might print J on the external terminal.

I was one of those who purchased the TI single board computer, and the memory expansion module put out by George Goode and Associates. I also bought the full complement of RAMs and EPROMs.

Other neophytes should stay away from "single board computers" until they have had a practical demonstration that the board will in fact do what the coloured pictures say that the board will do. They should start with some of the Radio Shack equipment or wait for the Japanese to come out with equipment which is fully supported with software. The high technology USA firms appear to have difficulty communicating with the common herd.

Percy Buzza
Dartmouth, Nova Scotia

New Improved Rat

Brian McCarson's program Rats (April 1981, p. 84) is a very interesting one which my family has had a lot of fun with. The article doesn't mention, however, that the program is designed to run with the RAM-based high-resolution subroutines which were supplied with earlier Apple IIs. Later versions, like ours, contain the Programmer's Aid ROM. I found this out the hard way when, after my wife and son typed in the program, the first run produced a Monitor Error.

The new lines to make the program run with the ROM-based routines are shown in Listing 1.

Richard I. Marmon
Papillion, NE

Born to Compute

Congratulations, you have succeeded in making my life miserable! A full three

weeks of the month are spent by the mailbox anxiously awaiting the next issue of *Microcomputing*.

By the way, I now have competition for time with your magazine in my own home. My two-year-old finds the pictures in *Microcomputing* fascinating and often tries to swap me even-up for her storybooks. Enclosed is an unposed snapshot of what may be your youngest fan "in the act."

Michael Amos
Mineral Wells, WV

If you want, we'll send her a copy of our writer's guidelines.—Eds.

First the Good News . . .

Soft Sector Marketing, which sells the Lazy Writer word processing program for the TRS-80 Model I, is a prime example of what good service should be. Not only have they been helpful in answering my questions concerning their software, but they sell excellent programs. The Lazy Writer word processing program is considerably better than any of the other programs offered for the TRS-80. It actually puts The Electric Pencil and Scripsit to shame.

All the features you wished you had in

(continued on page 211)

```
220  X = VX + DX(N):HX = X:HY = YU(N)
230  HY = YD(N):CALL LINE
235  X = VX - DX(N):HX = X:CALL LINE
240  HY = YU(N):CALL LINE
245  X = VX + DX(N):HX = X:CALL LINE
325  HX = X:HY = YU(N)
335  X = VX + RL*DX(N):HX = X
345  HY = YD(N):CALL LINE
350  X = VX + RL*DX(N - 1):HX = X
420  X = VX + RL*DX(N - 1):HX = X
425  HY = YU(N - 1):CALL POSN
430  X = VX + RL*DX(N):HX = X
435  HY = YU(N):CALL LINE
440  HY = YD(N):CALL LINE
445  X = VX + RL*DX(N - 1):HX = X
450  HY = YD(N - 1):CALL LINE
460  HY = YU(N - 1)
810  HX = HY = HCLR:LINE = - 11500:POSN = - 11527:INIT = - 12288
830  HCLR = 7:REM WHITE
3200 CALL INIT:HCLR = 255
3231 HX = 1 + HZ:HY = 1 + VZ:CALL POSN
3232 HY = MV*VZ + 1:CALL LINE
3300 HX = X:HY = Y:CALL POSN
3330 HX = X - HZ:CALL LINE
3410 HX = X:HY = Y:CALL POSN
3450 HY = Y - VZ:CALL LINE
3540 HX = X + 1:HY = Y + 1:CALL POSN
3545 HX = X - HZ + 2:HY = Y - VZ + 2:CALL LINE
3550 HY = Y:CALL POSN
3555 HX = X + 1:HY = Y - VZ + 2:CALL LINE
6075 HX = X:HY = Y:CALL POSN
6275 HX = X:HY = Y:CALL LINE
```

Listing 1.

Editor's Corner

Teaching Our Kids

By Harold Nelson
Microcomputing Technical Editor

If you've had a chance to watch children work with computers, you're aware of the energy and enthusiasm with which they engage their work. Children, it seems, can't wait to get their hands on the computer.

The real challenge to educators planning computer courses is not how to interest students in the computer. It's preserving students' initial interest while, at the same time, having the computer serve them as a tool for growth and development.

NECC 1981

The 1981 National Educational Computing Conference at North Texas State University last June brought a wide range of teachers, researchers and industry representatives together to consider a variety of issues involving education and computers.

Day-long preconference workshops included "Educational Computing for Administrators" conducted by Arthur Luerhman, "Computer Based Learning" by Alfred Bork and "Logo" by Dan Watt.

The three-day conference itself offered a total of 52 sessions on topics ranging from preschool to graduate school uses of computers. There were tutorial sessions on programming languages such as Pascal and Ada, and sessions conducted by representatives from the microcomputer industry.

Conference organizers were pleased by the numbers attending the conference and by the number of vendors and exhibitors showing hardware and software products. It was interesting to note that while mainframe manufacturers (Digital, Honeywell and IBM) were present and handing out literature, several microcomputer producers (Apple, Atari, Commodore, Radio Shack and Texas Instruments) stole the show with hands-on exhibits.

In fact, the "Computing" in the conference's title could well have been changed to "Microcomputing." Discussions on educational uses of personal computers predominated, both in and out of the formal sessions. It reflected the shift from large timesharing systems toward small microcomputer systems, even at the university level.

J. D. Spain of Michigan Technological University justified his move in this direction by pointing out how microcomputers can overcome the delays, communications breakdowns, access problems and student anxieties present when a timesharing system is used.

Some speakers talked about the flexibility of the smaller machines, while others discussed their economic soundness.

The Great Debates

Two areas of contention were in sharp focus at the conference. First was the clear antagonism between the educational community and software producers, large and small. Educators are suspicious of the producers' profit motives. The producers, on the other hand, believe that profit is an indispensible ingredient in the production of quality educational software. (See "Educational Computing—The Giant Awakes" by Lloyd Prentice on page 86.)

Some large publishers are considering the market. Others are currently developing packages. A few, such as Science Research Associates (SRA), have already jumped into the market. (Next month we will publish an account of MicroSIFT, which is attempting to remedy educational software evaluation problems for both existing and new products.)

The second and perhaps more profound issue is the debate among educators themselves over how best to use computers in the learning process. The debate among the various factions is likely to continue for some time (which also makes it difficult for software producers to decide what to do). Fine accounts of this debate (with some positive suggestions) are given in the articles "Through a New Looking Glass" by Henry Olds on page 62 and "Logo and the Great Debate" by Rick Carter on page 48.

Logo

The article on page 42 by teachers

of the Lamplighter School was originally a team presentation at NECC 1981.

This piece gives a real feel for the Logo language and some of the results that can be achieved. A similar project has been carried out in some New York City schools this past school year. And even though the environments are quite different, the results, in terms of student learning and enthusiasm, are very similar.

In fact, a teacher in the Bronx reports that one student, diagnosed as having a moderate learning disability, showed no symptoms of this disability when using Logo.

An introduction to the use of Logo with the learning disabled and physically handicapped is offered by Dr. Sylvia Weir of the MIT Logo Project in "Logo and the Exceptional Child" on page 76. Her work at the Cotting School for Handicapped Children in Boston is the basis of this article.

Plea to Teachers

There is a recurrent plea in several of the following articles and it is simply stated: *Teachers, get involved!* Teachers are encouraged to learn what is available and possible, to become the shapers and leaders of what seems to be an inevitable educational revolution.■

Sources of Information

The following list is provided for those wishing more information on topics presented in the following articles.

Books

Papert, Seymour, *Mindstorms: Children, Computers, and Powerful Ideas.* Basic Books: New York, 1980.

Taylor, Robert P. (ed.), *The Computer in the School: Tutor, Tool, Tutee.* Teachers College Press: New York, 1980.

Articles and Papers

Nelson, Harold, "Logo for Personal Computers." *Byte*, June 1981.

Papert, S., Watt, D., and Weir, S., "Final Report on the Brookline Logo Project." Logo Memos 53 and 54, MIT, Cambridge, MA, 1979.

Weir, S., "The Evaluation and Cultivation of Spatial and Linguistic Abilities in Individuals with Cerebral Palsy." AI Memo 570, MIT, Cambridge, MA, 1979.

-----, and Emanuel, R., "Using Logo to Catalyse Communication in an Autistic Child." D.A.I. Research Report 15, Department of Artificial Intelligence, Edinburgh University.

-----, and Watt, D., "Logo: A Computer Environment for Learning-Disabled Students." *The Computing Teacher,* vol. 8 (no. 5), 1981.

Periodicals

Educational Technology Center Newsletter, University of California, Irvine, CA 92717.

Classroom Computer News, PO Box 266, Cambridge, MA 02138.

ComputerTown, USA! News Bulletin, PO Box E, Menlo Park, CA 94025.

The Computing Teacher, Computing Center, Eastern Oregon State College, LaGrande, OR 97860.

Logo Systems

Texas Instruments, Customer Relations, PO Box 53, Lubbock, TX 79408.

MIT, AI Lab, Logo Group, 545 Technology Square, Cambridge, MA 02139.

✓See List of Advertisers on page 210

Are computers the answer to the declining quality of instruction as class size becomes larger?

Classroom of the Future

By Jeff Nilson

In the thinking about using computers in schools, what is remarkable is the extent to which people have ignored the reality symbolized by the election of Ronald Reagan as President. To many taxpayers, schools have become too expensive. At the same time, teachers' salaries have been declining by three to seven percent in real terms since the late 1970s. Teaching has become a job increasingly filled with stress and nonteaching work: filling out forms, organizing materials, correcting papers and maintaining student records. Curiously, the costs for schools are up while the rewards for teaching are down.

The response to what are viewed as runaway educational costs in many communities has been to hold the line on teachers' salaries at about half the inflation rate and to increase class size. Many systems have managed to enlarge classes by not replacing teachers who have left their systems for one reason or another. But in states like Massachusetts, tax cutting laws have sent 20 to 30 percent of teaching staffs looking for another profession or headed south in search of classrooms. The effect has been to increase teachers' nonteaching duties even more.

Within this climate, people concerned about the declining quality of instruction as classes grow larger may soon begin to ask if computers can reduce costs and increase efficiency as they have done in industry. In American industry where productivity has declined sharply over the past decade, a floodtide of companies has begun to explore replacing workers with robots. And so in education in the next ten or 15 years, economic and demographic forces may compel schools to try new ways of organizing classes using master teachers, aides and computers.

Within that time school systems may test the notion that remedial students should spend 60 percent of their time working on networked computing machines of some kind, and that because of the computers, remedial teachers should have a student load two or three times what it is today. Or schools may look at what kind of cost savings would accrue if a single master teacher, an aide and eight networked microcomputers taught a class of 55 or 60 students.

Such scenarios tap a fear that has grown up with the thousands of computers that have appeared in the schools: will computers by 1988 or by 1990 do *some* jobs as well as teachers do now and at less cost? In effect, will teachers be replaced by computers just as assembly line workers have been replaced by robots?

A Hypothetical Situation

To explore the answers to these questions, imagine a schedule of five eighth grade English classes of 58 students taught by a master teacher, one aide and eight microcomputers. Period one; Monday; a class full of 13-and 14-year-olds, who used to like school and still do pretty well in their studies, but who like being in love better, whenever they can work out all the details.

Twenty-eight students watch a video tape in one of the classrooms that house the double class. In the film, a girl must choose between her best friend and the group of kids that she has been trying to become part of for years. The group wants her to drop her friend because she isn't "cool" enough for them. In the open door between the rooms, the teacher talks to three boys about an ad they have written for the class newspaper. When they've finished writing a second draft, the boys will design the ad on one of the computers in the computer room; and then they will store the ad on a floppy disk along with the other parts of their groups' edition. Four students wait to do a reading comprehension tutorial on the two computers near the windows.

A cheer goes up from the six students sailing one of Columbus's ships

Jeff Nilson (95 Parallel St., Harwich, MA 02645) spent 14 years teaching middle school language arts in Bethesda, MD, Bedford-Stuyvesant in Brooklyn and in Dennis, MA. He is writing his doctoral dissertation in education at Boston University and is an editor at Classroom Computer News *in Cambridge, MA.*

across the Atlantic with a simulation running on another computer. They have just avoided a hurricane by slipping into one of the Atlantic currents that they correctly figured was close to their ship's position. After they finish their work with the simulation, the sailors will write about their victory over the storm in their versions of the ship's log.

Three other students puzzle over a punctuation exercise on another computer. They move the cursor over the extra comma and take it out. When they have gotten seven straight items correct, they will edit a paragraph on the computer, which the teacher has rewritten adding comma mistakes like the ones they are studying. If they forget the rule they are practicing, they can call up a short review lesson.

When the video tape is finished, the 28 students break into four groups of seven to think of reasons the young woman in the program should stay with her old friend or not. The teacher joins each group for a while to focus the discussion or to answer questions. From the lists of reasons, they will write the first draft of their arguments. Then they will use one computer's word processor and special thesaurus of strong nouns and verbs to rewrite their drafts. Finally, they will save their final drafts on disk for their classmates to load into one of the computers to read and comment upon.

And so the class goes through the year. The teacher works with groups of various sizes. Students talk about literature, work through prewriting activities, ask questions about some mechanics or usage rules they don't understand, edit pieces of student writing and so on. They spend about 40 percent of their time working with the computers: this includes taking tests, doing grammar, spelling, usage, and mechanics lessons and exercises, playing vocabulary games, rewriting compositions and commenting on classmates' writing, and working with simulations.

The teacher's aide answers the students' questions about the computer work and coordinates all of the record keeping for each student. Most of the teacher's preparation time is spent reading students' writing, studying their records, deciding what skills they need to work on next, and what instructional materials they should use.

It is unlikely that such classrooms will be established in the near future. The kind of software needed to run computerized classes has only now begun to appear. And the class organization implied by the description requires teachers who have received training in instructional media and design. Such teachers are rare in the public schools. When publishers have produced enough good language arts software and teachers have learned how to work with computers, video equipment and other media, school systems can begin to experiment with computerized classes. When they do, they will find that computerized double classes taught by a teacher and an aide who supervises the computer room could save from 15–20 percent of the cost for teachers over a five-year period. Assuming some of the computers do not melt during June of the fifth year, the savings would be even greater during the second five-year span.

In today's dollars, two teachers (each with eight years of experience), at $18,000 per year, would cost a school district $180,000 over a five-year period. On the other hand, one such teacher and an aide, at $8000 per year, would cost $130,000 over the same period. Add $20,000 for eight computers with disk drives, software and maintenance. Total five-year cost for the computerized class—$150,000. The $30,000 difference is 1/6 of the salary costs of the conventional classes, a savings of a little more than 16 percent.

For school systems even to be in a position to try out such a classroom organization by 1985 or 1986, three changes will have to occur in the relationship among school systems, teachers and the educational computing business.

Better Software

The first change involves software. Whereas there is now scarcely any software available to English language arts teachers, by the middle of the decade, teachers must be able to chose from hundreds of high-quality programs. Today, the software on the market is a mix of low-budget cottage industry programs, medium-budget material often produced by media distributors, and the first trial balloons floated by big publishers.

At best, the programs take no risks, and use the computer as a Skinner box. Much of this software comes from the pens of "instructional designers," people trained in learning theory and behaviorism. Much of the other software is poorly conceived, instructionally naive and sometimes pedagogically unsound.

For example, misspelled words dot one college board preparation program that requires students to pick the correct synonyms for words like "nefarious" in fewer than eight seconds. The program confuses teaching with testing by forgetting that limiting the time students spend learning something will likely keep them from learning it. In a reading comprehension program, students are not allowed to see the paragraph they just read, so they can use the text to prove their answers. Again the authors have ignored the difference between having students practice a skill and testing to see if students know the skill.

But what is most remarkable about current language arts software is the absence of programs that enable students to refine fundamental language skills. There are no word processors for students that make it easy to correct spelling errors in a composition or find ten strong synonyms for *run*, for example. Programs that purport to drill traditional grammar fail to build into the programs systems for determining when items should be reintroduced, procedures that were described in papers written by Marty Siegel's group at the University of Illinois several years ago. Though grammar drill programs may help students identify nouns and adjectives, they do not give students a chance to explore either the way English works or the reasons people write grammars of a language in the first place.

Yet for more than a decade artificial intelligence researchers have been developing techniques which could enable students to write their own grammars of English on the computer and then have the computer generate sentences based on that grammar. In vocabulary programs, students are supposed to learn thousands of words by picking the correct synonym or antonym from groups of four words. However, research indicates that at the very least words should be presented with their correct synonyms a couple of times before they are tested in a multiple choice format. Students learn words best when they are introduced in a defining context. Learning about word histories and prefixes, suffixes and roots helps, too. No vocabulary program on the market blends these techniques in a coherent system for teaching words and their meanings.

One important reason for the lack of language arts software is that publishers have not invested large amounts of money in software development as yet. Language arts teachers will be the last to adopt computers, the argument goes, because they are word people. And *bon mots* and bytes don't mix. Many publishers have viewed producing software as a risky business. According to one publisher's representative, the chance is still too great that a school system will buy one disk and make copies for all its schools. That's not a problem with books because it is more expensive to photocopy 100 books than to buy them.

Another editor, whose company has sold a considerable amount of math software, believes math teachers are easier markets to sell to than English language arts teachers. "After all," he says, "there's basically only one way to teach math. But there are so many philosophies of reading. And if you produce software with one philosophy, people with another way of seeing things won't buy from you."

"Computer software is a half-life product," observes a product designer from another major publisher. "Books have a life expectancy of five years; software, half that."

That fact plus piracy problems and high development costs make it difficult for computer advocates within publishing companies to justify the investment of hundreds of thousands of dollars on software. With the coming of computers, publishing committees, vice presidents, senior editors and the like—some of whom were once salespeople—have to decide between investing in texts, which they

know about, and software, which they don't. This has caused several large educational publishers to stand on the sidelines while watching how much money other publishers make. But there are indications that this is changing.

Several publishers will soon be developing reading and language arts software, according to sources within these companies. New reading programs will be coming out during the 1981–82 school year to compete with materials that have already been produced by companies like Scott-Foresman and SRA.

In addition, the federal government will be spending several million dollars to develop and test the effectiveness of microcomputer software to teach writing and reading skills. Thus perhaps by 1985, there will be enough computer software to organize computerized classes.

Teachers and Jobs

The second change that must occur before schools are ready to try computerized classes has to do with teachers and jobs. Teachers will never be fired to be replaced by computers. The public school system, which is run by people who are products of it, guards the traditional security of teachers' jobs. Unless there are mandated cuts in school funding, it is virtually impossible to fire teachers for any reason. For schools to use computers to teach large numbers of children, school administrators will be limited to filling teaching vacancies with computers. Because of that, predictions of a teacher shortage by the mid-1980s will have to come true.

In the early 1960s the post-war baby boom caused a massive need for teachers. But if there is a shortage in 1985, it will be because teaching is one of the least attractive occupations for many college students. Already there is a nationwide shortage of math teachers, even in states like Massachusetts, where large numbers of teachers have been laid off. At Boston University's School of Education, where future math and science teachers once filled lecture halls, there were almost no students taking practice teaching in math or science education during the last academic year.

In addition, the number of women going into teaching has declined sharply, according to Robert Wood, who is coordinating National Institute of Education research on the state of teaching in America. Women are going into other professions because of increased opportunities, he says. And this is depleting the ranks of the most talented group of teachers in education.

What will teaching staffs be like in 1985? Most teachers will be in their 40s and 50s, with fewer and fewer young teachers joining their ranks each year. Why should young people go into teaching? Today, teachers in their 30s are trading better money, less stress and more intellectual challenge for their long summer vacations and their two hours of correcting each night. One former teacher, for example, who now shows customers and salespeople how to use a big manufacturer's computers, reports being less tired and more excited about his job despite working from 9 AM to 5 PM each day, 49 weeks a year. With faculty salaries losing the race with inflation, teaching has again become one of the poorest paid professions.

See List of Advertisers on page 210

Efficiency

The third change involves recognizing that having computers handle a significant amount of basic-skills instruction will be more efficient than having teachers manage that instruction. Today, many people do not recognize that computers can teach *word* skills. The thought rekindles the conflict between humanism and efficiency, between arts and machines. To use the machines to teach arithmetic is one thing; but Keats? Isn't teaching students to read and write their language one of the few jobs in a mechanized world left for humanists? Many English and reading teachers see themselves as faithfully guarding the tradition of Shakespeare and Shelley, standing on the bastions of fortress English repelling armies of linguistic barbarians. How could they betray this trust in deference to machine efficiency?

The fact is that most language arts teachers, whether they teach reading or English, spend precious little time actually teaching writing and reading skills. Instead, they lose countless hours organizing and correcting the paperwork they must assign to have their students practice the very skills they are teaching. Twenty or 30 hours are often required for correcting each week, writes Janet Emig in the *English Journal:* 75 minutes to correct and record the scores from 125 spelling tests at 45 seconds per test; six or seven hours to read and comment on 125 compositions at three or four minutes per composition; and so on.

In a study done for the Center for the Study of Reading at the University of Illinois, Dolores Durkin found that reading teachers spent almost no time actually teaching comprehension. Rather, they spent 40 percent of their time giving out assignments and checking to see if students understood them. Another 22 percent of the class period was devoted to getting children the right materials to work with, gathering groups together or reprimanding children. During the rest of the time, children worked mostly on other reading skills. Whatever teaching was done consisted mostly of *mentioning*, where the teacher defined something and perhaps gave an example of it before defining something else.

Another study done at the University of Florida showed that only about 40 percent of the high school students in the U.S. write a composition at least once a week. At the same time one-fourth of the students wrote only once or twice a semester. If students learn to write by writing, what is the effect of this lack of composition work?

Conclusion

Good teaching is a paradox. The more teachers individualize instruction, the less time they can work with students because of all the recordkeeping involved. The more hours they spend on administrative tasks, the less time they have to prepare lessons. The more papers they assign, the more papers they have to correct, and the longer the period of time between the students' doing work and finding out how many mistakes they made.

Computers could break this paradox during the 1980s. Will they? I refer you to a corollary of Murphy's Law: when an event is predicted, it changes simply because you've predicted it. ■

Learning with Logo At the Lamplighter School

By six Lamplighter teachers

All students at the Lamplighter School, even three-year-old preschoolers, enjoy daily use of a TI 99/4 computer with TI Logo. Computer use is only one among many interesting daily learning activities. Drawing pictures with the computer does not replace working with paper, paint and scissors in art class. Discovering geometric concepts with the Logo turtle does not replace multiplication in math class. Still, Logo has had a profound effect on both teachers and students.

Introduction
By Theresa Overall

The Lamplighter School in Dallas is a private school consisting of a preschool and grades one through four. Our students come from a variety of backgrounds and represent a wide range of learning abilities.

Students daily engage in a variety of learning activities, including use of a microcomputer.

But at Lamplighter we don't have computer instructors with masters degrees in computer science. We don't use strange computer jargon. We don't teach our students in a computer lab down the hall. All teachers teach Logo on TI 99/4 personal computers in their own rooms.

That is one unique feature of the Lamplighter Logo project; teachers are teaching with Logo.

The way we accomplished this was that a core group of teachers began learning Logo in the fall of 1978. During the spring of 1979 we *played* with Logo, developed some different teaching techniques, got more confidence in our Logo skills, and started teaching some third graders. We learned a lot, taught ourselves more Logo, and revamped some of our techniques. We taught a few more teachers, forming a larger steering committee of at least one teacher from each grade level. Then each member taught her fellow team members. We've had staff meetings during the year for the entire faculty to learn Logo, write Logo procedures, find out what other teams are doing in their classrooms, discuss different teaching techniques and grow in personal Logo skills.

None of us were computer specialists when we started. In fact, every one of our teachers could share a horror story or two about his own work with Logo and about sharing Logo with his students. But there would also be many tales of triumph. It's fun and really easy to use. And all Logo users at Lamplighter, both students and teachers, are learning.

The Lamplighter Philosophy
By Pat Lola

I would like to introduce you to Lamplighter's philosophy of educating young children. This philosophy

At Lamplighter no child is always first,
and consequently,
no child is always last.

is the foundation of our computer program.

The school was founded by Natalie Murray and Sandy Swain 28 years ago in a red brick farmhouse on five acres of land that were, at that time, beyond the Dallas city limits. That school, as is the case with the present school, gave no report cards, rang no bells, and allowed children to learn at their own rates, competing only with themselves.

I do not mean to imply that all the 450 children at Lamplighter work separately or individually all the time. What a lonely road that would be. Children work in groups, some large, some small, sometimes with a teacher or perhaps sharing with another child. But the feeling that pervades the entire system is that each child is unique and individual. Children are not forced through the same work book, the same reader or the same computer procedures. At Lamplighter no child is always first, and consequently, no child is always last.

The founders, and the present directors, feel that success breeds success. The purpose of the school is to ensure that, through high academic standards, the children will develop skills that will help them cope successfully with the world in which they will live—a world we cannot predict. Therefore, the concern is *with teaching* a way to think and, at the same time, teaching the child to believe in himself.

Following this philosophy, we use the computers to help the children develop logical thinking skills while working on projects of their own design. In many schools computers are used only to aid in teaching a given curriculum. But Logo is designed to allow each child to manipulate the system according to his developmental skills and needs at a given time. Because there is no way to exhaust Logo's potential, there is no artificial limit on what the child can do with the system.

Surely the computer is the ultimate individualized learning tool, because it can go in any direction, at any pace and to whatever depth the child wants to take it.

Eric Jonsson (founder of Texas Instruments and former Dallas mayor) has been a long-time friend of Lamplighter. In 1969 he was the chief financial underwriter for the expansion of the school to its present 1.5 million dollar building. He has contributed not only financially to the school, but through his continual guidance of the Board of Trustees as well. Six years ago he and Texas Instruments helped to implement one of the first calculator programs for young children. It is still an integral part of our math program. In fact, the children have free access to many Little Professors, Speak & Spells, Speak & Reads and other electronic learning aids. The school has a history of successfully using this type of electronic technology.

The next step was computers—computers, of course, within the framework of the purposes and philosophy of the school. So in 1978 Mr. Jonsson funded the development of a computer system that could be controlled and manipulated by the young child. This features Logo, a graphics-oriented computer language, developed at the Massachusetts Institute of Technology and implemented by Texas Instruments. We now have 50 TI 99/4 computers at Lamplighter, used in a variety of ways by three-year-olds through fourth graders—all feeling successful at their own levels.

Growing with Logo

By Kay Murphy

Preschoolers at Lamplighter are introduced to Logo via four procedures written by Lamplighter teacher Coleta Lewis. Her procedures act like a simplified version of Logo itself—a feature that seems to be unique to Logo. These procedures allow preschoolers to experiment with concepts of shape, position, color, direction and speed. Children do this by controlling the computer with single-key commands. Every aspect of these procedures is in harmony with Lamplighter learning goals and philosophy.

The carefully thought-out concepts introduced to students in the preschool overflow into the primary grades. Logo allows the student to attain a thorough understanding of these concepts, not just rote application of them.

In the primary grades we introduce primitive Logo graphics commands to students. These allow children using the computer to gain further control of the sprites and the turtle.

We take into consideration the learning styles of the children. In the initial introduction to turtle graphics, for example, we might instruct children by using the whole body and actually playing turtle. Through careful questioning we can solicit insights in-

to concepts about direction, change of direction, movement forward and backward and distance.

We might also use a triangular card with a pencil placed through it. Logo commands such as PENUP, PENDOWN, FORWARD, BACK, RIGHT90, etc., can be executed and experimented with on paper.

Moving from the known to the unknown, we would then proceed to the keyboard. Now the child can apply the primitive commands he or she has learned to "teach" the computer to produce his design, pattern or random display.

While having tremendous fun a child might, for example, draw a polygon, discovering for perhaps the first time that a square has four equal sides and that it took four equal turtle turns to make the corners. Similarly, another child might discover a triangle and a rectangle. Someone else might put these together to form a house.

All this time, the child is learning at his or her own conceptual level, progressing at a rate commensurate with his or her understanding. When the indicators are given (by the child) for more advanced turtle geometry, further concepts are introduced, such as inputs, variables, etc.

Sharing Ideas
By Mitzi Mckool Dafoe

One benefit of having Logo in the classroom is that it gives students an opportunity to share ideas easily. They are excited about the procedures their classmates have written. And they eagerly explain their own procedures to one another.

Fig. 1.

It is not a competitive atmosphere but rather one in which they actively help each other to learn how the computer works. When one child teaches another, he also learns. This develops their communication skills. In the course of explaining a procedure, the student has to analyze his own method of program writing, organize his thoughts and put them into words.

Many times a child will copy down someone else's procedure, and type it into the computer to see how it works. But he doesn't stop there. Inevitably, the child will edit and change the procedure to fit his needs and wants. For example, a second grader came to school one day with a procedure that his fourth grade brother had given him. He excitedly typed it in to see what it did. The video screen showed exploding balls, all of the same color. He was able to figure out how the procedure worked and later altered it by changing the colors of the balls and their speed. Other children joined in and made their own changes.

I've seen students spontaneously work together on a joint computer project. The end result was achieved through discovery—they didn't have a set idea in mind when they started. One such group was working at placing different colored balls in black squares and then placing these at various locations on the screen. When three boxes with colored balls appeared in a column, someone suggested that if the balls were red, yellow and green it would look like a stoplight. They did this and added a stand to produce their stoplight (see Fig. 1). By the end of school that day every second grader in the school knew how to make a stoplight. It was not long before some started adding trucks, planes and trees around the stoplight to make a complete scene.

This is an example of how students build on one another's ideas. They were solving problems together—problems that they had discovered themselves and wanted to solve themselves. These problems involved positioning shapes, using color and relative speeds (e.g., a plane moves faster than a truck).

With Logo, students see themselves and the worth of their classmates in new and different ways while learning to develop and manipulate new concepts.

Children Teach the Computers
By Jalna Housey

Logo provides an atmosphere in which the child can determine his or her own level of development while learning about computers and computing.

The child is told that the computer needs him. The student becomes the "teacher." The child then comes to realize that the computer does not know *everything*. An example of the use of this knowledge came to my attention through a creative writing experience.

The children had been working on computers for quite some time. In one of their creative language classes they were asked to write a story entitled "The Computer That Could Do Anything." The children wrote wonderful stories about computers that would make beds, do homework and even prepare peanut butter and jelly sandwiches for lunch. But one child's


Take Your Cue From Vista . . .
When it comes to Apples, take your cue from Vista's A800 Eight-Inch Floppy Disk Controller. The A800 offers a cost-efficient approach to software compatible disk memory expansion for your Apple II® computer. The A800 Controller enables Apple II users to access up to five megabytes of online storage through conventional disk operating (DOS) commands.
The Control and DMA Logic provides high speed (1 microsecond per byte) transfer of data from the disk drive directly to the Apple II memory without processor intervention. Plus, the Phase-Locked Loop Data Separator provides the ultimate in data reliability.
AND THERE'S MORE.
The controller is compatible with the most popular disk operating systems for the
It's a Sure Shot.
Apple II computer. It also interfaces to all Shugart/ANSI Standard Eight-Inch Floppy Disk Drives. The A800 provides complete IBM format compatibility in both single and double-density modes.
THE HOT SHOT AT A COOL PRICE.
The A800 is reasonably priced at $595. A price you can't beat when you compare its quality and performance.
CALL YOUR SHOTS.
Vista offers you a complete line of peripheral equipment to maximize the capabilities of your Apple system including: High capacity Mini-floppy Disk Drives, the Vista Music Machine 9 and the Vista Model 150 Type Ahead Buffer. In addition, Vista offers a line of advanced components fully compatible with the TRS-80* and S-100 based computers.
Vista Computer Company
1317 E. Edinger Avenue
Santa Ana, CA 92705
(714) 953-0523
Call Toll Free 800-854-8017
® Apple II is a registered trademark of Apple Computers, Inc.
* TRS-80 is a trademark of Radio Shack, A Tandy Co.

story described a computer that she would teach to add 2+2 and get the answer 4, but wouldn't teach it the answer to 4+4. She realized she had the power to withhold that information from the computer.

In describing Logo, we must say that it is an understandable language, using English words familiar to children. It is a language in which concepts can be introduced, understood and used from the basis of the child's understanding. Using the child's conceptual knowledge, it allows the student to create his or her own unique procedures rather than having to simply respond to the demands of the computer. The child is actually teaching the computer rather than being taught by it.

Powerful Extras

By Sheila Leventhal

One aspect of Logo that we find very exciting is how beautifully it embodies the Lamplighter philosophy of individualization. Each child can create something beautiful on the screen and something that is uniquely his own.

We continue to be surprised and very enthusiastic as we watch our children operate on varying levels of expertise, and yet with cooperation, communication and appreciation. They interact with one another constantly in a spirit of togetherness.

It seems that regardless of the level at which a child is working, from preprogramming to using superprocedures, he can feel success and a very real sense of accomplishment.

The children truly never seem to notice or even concern themselves with who is working at what level. Skill levels seem to *lack significance* with Logo.

A child operating at the pre-programming level can cause quite a flurry of interest among his peers by creating a simple line design or forming a new shape. On numerous occasions children in the upper grades have observed procedures or shapes created by first graders or preschoolers and can't wait to try to duplicate it, modify it or simply play around with a new idea.

This, of course, provides excellent opportunities for success by the child who perhaps does not always feel successful in other areas—academic, social or whatever.

It has become obvious to us that it is not necessarily the most academically successful child who is most skillful with the computer.

We've seen phenomenal computer success in children who are artistically inclined, with children who particularly enjoy problem solving, with children who are somewhat intimidated with pencil and paper tasks, with children who may be having some difficulty with reading skills, with children whose reading skills are quite advanced and even with children who are not yet reading.

It is not necessarily
the most academically successful child
who is most skillful with the computer.

Another interesting phenomenon we have observed is that the children who are the leaders or the "experts" are always changing, unlike some academic situations where Johnny is known to be the math "whiz" and everyone just accepts it. One week the expert will be this child or this group of children and the next week it may be another, as each child explores the area or areas that appeal to him personally. For example, in my classroom, one rather quiet little girl, shortly after our introduction to turtle graphics, wrote a short procedure that produced fantastic results. For days her procedure was the rage of the fourth grade. Everyone copied, modified, manipulated or incorporated her procedure into his own.

Another child discovered that he could write conversation, post bulletins, etc., and this suggested dozens of variations to other members of the class.

One clever little fellow discovered how to erase characters from individual titles so that certain letters couldn't be used; and later began to change the letters so if you attempted to type a *q* you just might get an *s* instead. *They had a lot of fun with that discovery*. We could cite numerous examples of this occurring in our classrooms with many different children.

Individualization, promoting improvement in an individual's self-concept, in peer relationships, in ability to communicate . . . these are truly powerful, exciting *extras* of the Lamplighter Logo project.■

See List of Advertisers on page 210

The "back to basics" battle cry has been supplanted by the "ahead with Logo" banner hoisted by some educators.

Logo and the Great Debate

By Richard Carter

Educators are involved in a great debate about how computers should be used in our schools. Rather than take a particular side, I would like to suggest Logo as a potential peacemaker—a way to combine what may seem like opposing views.

The headlines tell us that a veritable microcomputer invasion is occurring in our schools. A recent study by the Minnesota Educational Computing Consortium found that there were 25,000 microcomputers in the nation's schools.

But before we declare a revolution we must note that 25,000 divided by the number of schools in the country is not all that much computing power; and although things are changing faster than many of us ever imagined, it looks as though computers—for the immediate future at least—are going to be a relatively scarce resource in school classrooms.

Given this reality, how should computers be dealt with by the schools? Here is where the debate begins. The question has already been addressed by many authors. In reviewing some of those answers, I find that most of them fall into the following categories:

1) Some argue that the computer should be used as a machine to teach traditional curricula. This use is commonly referred to as computer-aided instruction (CAI). People taking this position argue that our children need plain old basic skills, and the computer will help children master them; computers can and should be used to ensure that all children are literate in basic skills such as reading, writing and arithmetic.

2) Others argue that computers should be seen as a new social and technological phenomenon, and children should learn about them. These authors address such issues as what a computer is and the social impact of computer technology. They believe that children must be prepared to live in a computer-based society—they must be exposed to the new technology and the social and personal issues that it raises. One advantage of this focus is that computer topics can be inserted into standard curricula throughout the grades. Social issues can be addressed in social studies units, what a computer is can be dealt with in the early grades in science or math and things like programming can be taught as a math course. In the latter case, actual computer use might be postponed until the upper grades.

3) A third group argues that both of the above approaches miss the potential impact of the computer on children's learning. They argue that the power of the computer can be best exploited by teaching all children how to program. Members of this group suggest that programming can provide children with intellectual skills not available by any other means. Many people who advocate this position believe that the computer has the potential to transform learning—to teach these new skills in a revolutionary way. Judah Schwartz has called a computer used in this way an "intellectual amplifier." (See Refs. 7 and 8.)

Learned people make compelling arguments for each of these positions; and in the best of all possible worlds, we might try to implement them all. But given that the computer will, for the present, be a scarce resource, what are schools to do? If a classroom of children has access to a computer, how should it be used?

Perhaps there is a way to combine some of these goals. In fact, this is the suggestion made by John Levin in a recent study on the impact of computers on schools. Levin and his fellow researchers suggest children start by running programs that will help them master basic skills; at the same time, they can be introduced to programming by learning how to modify these programs. (See Ref. 4.) It is an exciting idea, but there is a problem. Most programs are written in computer languages that are not very easy for children to understand. In languages like BASIC it is not easy for a novice to get any sense of what is going on by looking directly at the code of a program. Levin suggests modifying programs by doing such things as changing the value of variables, but in my experience it is not often easy to see the role of a particular variable in the larger structure of most computer programs.

BASIC, though designed to be easier to deal with than some other languages, is still not very transparent to

Richard Carter (33 Sharon St., Medford, MA 02155) teaches courses in computer literacy and Logo at Lesley College in Boston, and is doing graduate work at MIT.

a beginner, and it isn't very easy to make it more transparent. Although Levin's idea is appealing, I fear current educational programs leave a large gap between this ideal and the current reality.

Bridging the Gap

But this is a gap that I believe the Logo language is beautifully designed to fill. Unlike BASIC, Logo is an extendable language. In languages like BASIC, you're limited to a certain set of primitive commands, and programs must be constructed using only those commands. In Logo a user can construct his own new set of primitive commands. Logo is really not one language, but a multitude of user-invented languages. This quality gives Logo several features that are difficult to achieve in a language like BASIC.

Before considering how this capacity can be used to create transparent programs that children can modify, let me try to clarify how Logo allows for user-invented languages.

One implementation of Logo is a geometry/graphics world in which a student can create pictures by moving a line-drawing graphics *turtle* around a video display screen. The user begins with a few primitive commands for controlling the turtle. For example, FORWARD "*some number*" will move the turtle forward that number of "steps," and RIGHT "*some number*" will rotate the turtle that number of degrees to the right. The user can use these commands to *create geometric figures*. For example:

```
FORWARD 40
RIGHT 90
FORWARD 40
RIGHT 90
FORWARD 40
RIGHT 90
```

This will cause the turtle to draw a square (see Fig. 1). Since Logo is an extendable language, a student can use these commands to define a new primitive command he might call "SQUARE":

```
TO SQUARE
   FORWARD 40
   RIGHT 90
   FORWARD 40
   RIGHT 90
   FORWARD 40
   RIGHT 90
   FORWARD 40
   RIGHT 90
END
```

Once he has done this, SQUARE becomes part of the computer's language; every time the student types the command SQUARE, the turtle will draw a square—in this case with a side of 40.

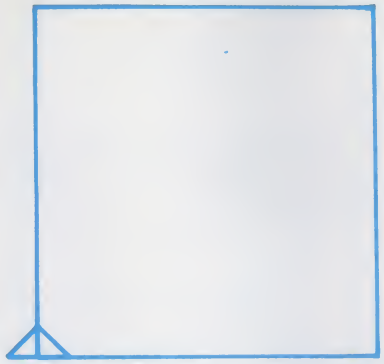


Fig. 1. Drawing a square with turtle graphics.

One example of Logo's flexibility is that you can do things like redefine commands to simplify the language. If, for instance, you feel that the use of numbers may be a complicating or overwhelming factor for a group of young children, it is easy to create a "mini-language" where you define the letter F as meaning FORWARD 10 and the letter R as RIGHT 30. Thus, when a child types an F, the turtle will go forward 10 steps. With these two single-letter commands, very young children can draw their own turtle graphics and even define new commands. After drawing a triangle with F's and R's, a child could name it, say, T; when she wants a triangle, all she has to do is type T. The child is not only working with geometric shapes, but is being introduced to powerful computing ideas, and is gaining background that will make the introduction of more complex ideas, such as numbers, easier to deal with.

In this way Logo can be used to create languages that are appropriately transparent for the needs of particular users. Of course, this extends beyond the needs of very young children or beginners. For example, some students may be working on a project in which they need to find the distance between two objects on the screen, but they may not be ready to handle the square roots necessary to find distances. In this case one can simply make a Distance command that will determine the distance between two points on the screen.

Of course, it is possible to create simplified sets of commands for children to manipulate in BASIC or even in a language like Pascal, but the *process* of doing this is complex, and that process is *not* available to the children who will use that mini-language. The power of Logo is that the same process used by the designer of a simplified language is immediately available to the children so they can begin to create and extend their own language (e.g., T, to make a triangle, can be a new command created by a child).

Although there is geometry in the above use of Logo, the emphasis seems to be on children learning to control the computer through programming. How can this capacity be applied to creating educational software that both has transparency and will allow meaningful modification? Let us take another example.

Another Example

Suppose I have some children in my classroom who are working on the relations between numbers 1 to 20. One way is to have them estimate distances up to 20. Suppose I have a computer game in which they have to *throw* something (by typing in a number on a keyboard) some distance (1

✓See List of Advertisers on page 210

to 20) to try to hit a randomly positioned target, and then if they miss, modify their guess (throw again) based on how close they came to the target.

To program such a game in Logo, I might create the following commands:

- PICK.DISTANCE (to choose a random distance 1–20)
- DRAW.TARGET :DISTANCE (to draw the target at the chosen distance)
- GET.GUESS (to get the player's guess)
- THROW :GUESS (to move the turtle toward the target the distance guessed by the player)
- CHECK.IF.HIT :GUESS :DISTANCE (to see if it's a hit—if the chosen distance and the player's guess are the same)
- TELL.IF.HIT (to tell the player what happened)

It is important to realize that the commands are not remarks or comments—each of them becomes a part of the language once a user defines it. Thus the command DRAW.TARGET if typed into the computer would cause the turtle to draw a target. One might define it as:

```
TO DRAW.TARGET
   FORWARD :DISTANCE
   SQUARE
   RIGHT 180
   FORWARD :DISTANCE
   RIGHT 180
END
```

The first FORWARD :DISTANCE command moves the turtle forward the distance chosen by PICK.DISTANCE. SQUARE1 causes the turtle to draw a square like the earlier SQUARE command, except scaled down appropriately in size. The last three commands move the turtle back to where he started (see Fig. 2). to create SETUP and PLAY:

```
TO SETUP
   PICK.DISTANCE
   DRAW.TARGET :DISTANCE
END

TO PLAY
   GET.GUESS
   THROW :GUESS
   CHECK.IF.HIT :GUESS :DISTANCE
   TELL.IF.HIT
   PLAY
END
```

and then combine these to make the GAME:

```
TO GAME
   SETUP
   PLAY
END
```

To play this game, the child types the command GAME, and the turtle first draws a square target between one and 20 steps away (PICK.DISTANCE and DRAW.TARGET). The child then types in a number (GET.GUESS), which moves the turtle toward the target (THROW). If he misses the target (CHECK.IF.HIT), he gets another chance (PLAY), and must increase or decrease his guess depending on where his turtle "throw" landed. When he hits the target the game ends (CHECK.IF.HIT).

Children enjoy this game and sharpen their sense of the relationships between the numbers 1 and 20 as they use it. But what about modification? Suppose two children in my class know their numbers well and want to expand the 1 to 20 range of the game up to 100, and further that they want to change the game so that they can be rocket ship captains trying to land on a space station.

Let's begin with changing the range. First, I can show them the commands of the game (Fig. 2), and explain that PICK.DISTANCE sets the distance that the target is away. I can then let them look inside this definition to see how the distance is randomly chosen.

It might have been defined as

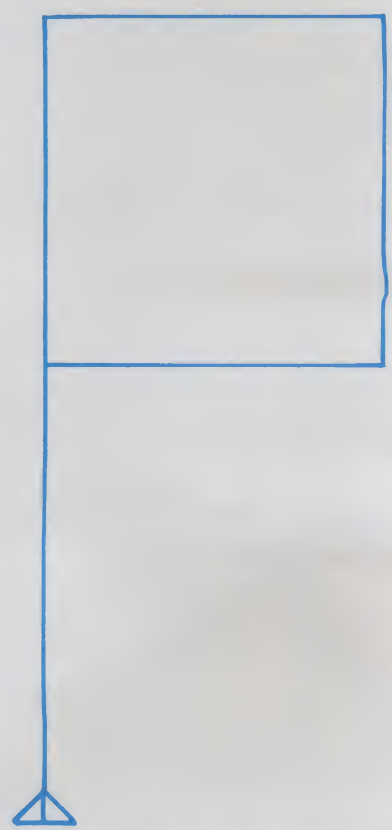


Fig. 2. The turtle draws a target.

follows:

```
TO PICK.DISTANCE
    MAKE "DISTANCE RANDOM 20
END
```

I would explain that PICK.DISTANCE chooses some DISTANCE up to 20, a different one every time. From here it is not hard to change RANDOM 20 to RANDOM 100 and help them make a new command that could be called PICK.DISTANCE.100 (Defined as: MAKE "DISTANCE RANDOM 100).

Once I've helped the children expand the range of the game, they next turn to making the target a space station. At this point the children might pick out themselves the fact that DRAW.TARGET draws the target, and if these children have used turtle graphics in exploring geometry, perhaps using the simplified commands described above, they would have a ready tool for creating their own drawing of a space station and could use it to replace my target. They might create a command called DRAW.STATION and simply insert it into the program. Their modified program might look like this:

```
GAME
    SETUP
        PICK.DISTANCE.100
        DRAW.STATION
    PLAY
        GET.GUESS
        THROW :GUESS
        CHECK.IF.HIT :GUESS :DISTANCE
        PLAY
END
```

Conclusions

These rather simple examples (geometry and numeration) are only two of the curriculum areas that have been explored using Logo. Others range from physics (see Ref. 2) to music (Ref. 1) to poetry and creative writing (Refs. 3 and 9). All of these applications can involve the same possibility of making transparent and modifiable environment/languages for children's learning.

In the past, using computers for CAI and teaching children to program computers have often been argued as competing approaches. The extendability of a language like Logo allows the possibility of combining these two approaches so that children can use programs for learning basic skills, while, through modifying and extending these programs, they can be introduced to fundamental ideas of computer programming.

One of the special bonuses of this approach is that it leads naturally into discussions with children about how they learn and what helps them learn. If several students are making changes in a program, it becomes natural to talk about what changes make learning harder or easier or more interesting. In this way you can help children begin to think about their own learning and how they learn best.

This use of computers presents the possibility of a school in which a creative computer culture can develop, where children are not only learning by using educational programs in various disciplines, but where these same children are learning to understand the structure of those programs and are able to modify them in ever more complex ways until they can create learning environments for each other and themselves.■

References

1. Bamberger, J. *Developing a Musical Ear: A New Experiment*. Logo Memo 6. Cambridge, MA: Massachusetts Institute of Technology. July 1972.
2. diSessa, A. "Learning Physics With Dynaturtle," In Papert et al. *Final Report of the Brookline Logo Project Part II: Project Summary and Data Analysis*. Logo Memo 53. Cambridge, MA: MIT. 1979.
3. Lawler, R. *One Child's Learning: Introducing Writing with a Computer*. SIGCUE Bulletin. July 1980.
4. Levin, J., Kareev, Y. *Personal Computers and Education: The Challenge to the Schools*. La Jolla, CA: Center for Human Information Processing (Univ. of CA, San Diego). 1980.
5. Microcomputing Report of the Minnesota Educational Computing Consortium 1979–1980 Laurendale, MN: MECC Instructional Services Division (2520 Broadway Drive). 1979.
6. O'Shea, T. "Current Work and Plans with LOGO." SIGCUE Bulletin V. 7 No. 4. 1977.
7. Schwartz, J., Olds, H., Willie, N. *People and Computers: Who Teaches Whom?* Newton, MA: Educational Development Center. 1980.
8. Taylor, R. *The Computer in the School: Tutor, Tool, Tutee*. New York: Teacher's College Press. 1980.
9. Watt, D. *Final Report on the Brookline Logo Project, Part III: Profiles of Individual Students' Work*. Logo Memo 54. Cambridge, MA: MIT. 1979.

✓See List of Advertisers on page 210

Whither Goes the Turtle?

By Joseph F. Rousseau and Stephen M. Smith

Although Seymour Papert has many intriguing ideas on revolutionizing education with microcomputers, enough questions emerge to put his theories on hold until they can be mulled over by more people than are currently familiar with his work.

This is not to suggest that years and years must pass before conclusive research data is in, but only that his ideas should be thoroughly examined in light of various learning theories, and not just the one that he mentions so frequently in *Mindstorms*.

It is understandable why Papert has based so many of his ideas on the thought of the Swiss educator and psychologist, Jean Piaget. Papert studied with Piaget for a number of years, and would be expected to use those theories he was most familiar with. Unfortunately, Papert seems to have forgotten some of the major premises of Piaget's theories about the development of learning.

Before looking at the differences between Papert and Piaget, perhaps a brief description of Piaget's theories about learning is in order.

Piaget in a Nutshell

The crux of Piagetan thought is that a child learns through experience, and develops a framework for dealing with his environment in relatively predictable stages. These stages are closely linked to chronological age.

A person matures intellectually as he gets older by coping with problems presented by his environment which becomes more sophisticated as his relationship to the environment becomes more complex.

Up to about the age of two the child explores the world through sensorimotor faculties. By coping with new objects and situations, the infant develops internal structures based upon his organization of all previous coping behaviors, thus making it easier for the child to handle an even greater variety of new situations.

Between the ages of two and seven, children are able to think beyond the immediate situation, but thinking still usually requires concrete objects or events for its initiation and guidance. The child remains oriented toward, and controlled by, whatever situation is at hand. Exposure to many different experiences during this stage increases the child's repertoire of possible responses to new problems that might arise.

Shortly after the age of seven the child begins to create new intellectual structures from existing internally organized knowledge. This lets him solve problems internally—through thought—rather than through actual manipulation of objects. Since new structures have emerged that give the child a very real power for thought, the child feels more and more intellectually competent, and is viewed as such by others. This is increasingly apparent as the child progresses through elementary school.

Near the end of junior high school, children shift to the type of thinking characteristically used by adults. This adult thought is independent of immediate situations. The child is now capable of the kinds of formal thought necessary for long sequences of reasoning. This logical thinking reflects the internal framework that has developed from the many experiences and problems in the child's environment.

Permeating this entire theory of intellectual development are two subprocesses that Piaget saw as necessary for learning how to think; *accommodation* and *assimilation*. Accommodation occurs when a child experiences something that doesn't fit existing intellectual structures, but makes it fit by changing his structures to include the experiences. Assimilation is simply incorporating the new experience into an existing structure. Obviously, experiences that trigger these may occur in a haphazard manner, or they may be presented in the school as part of an organized curriculum.

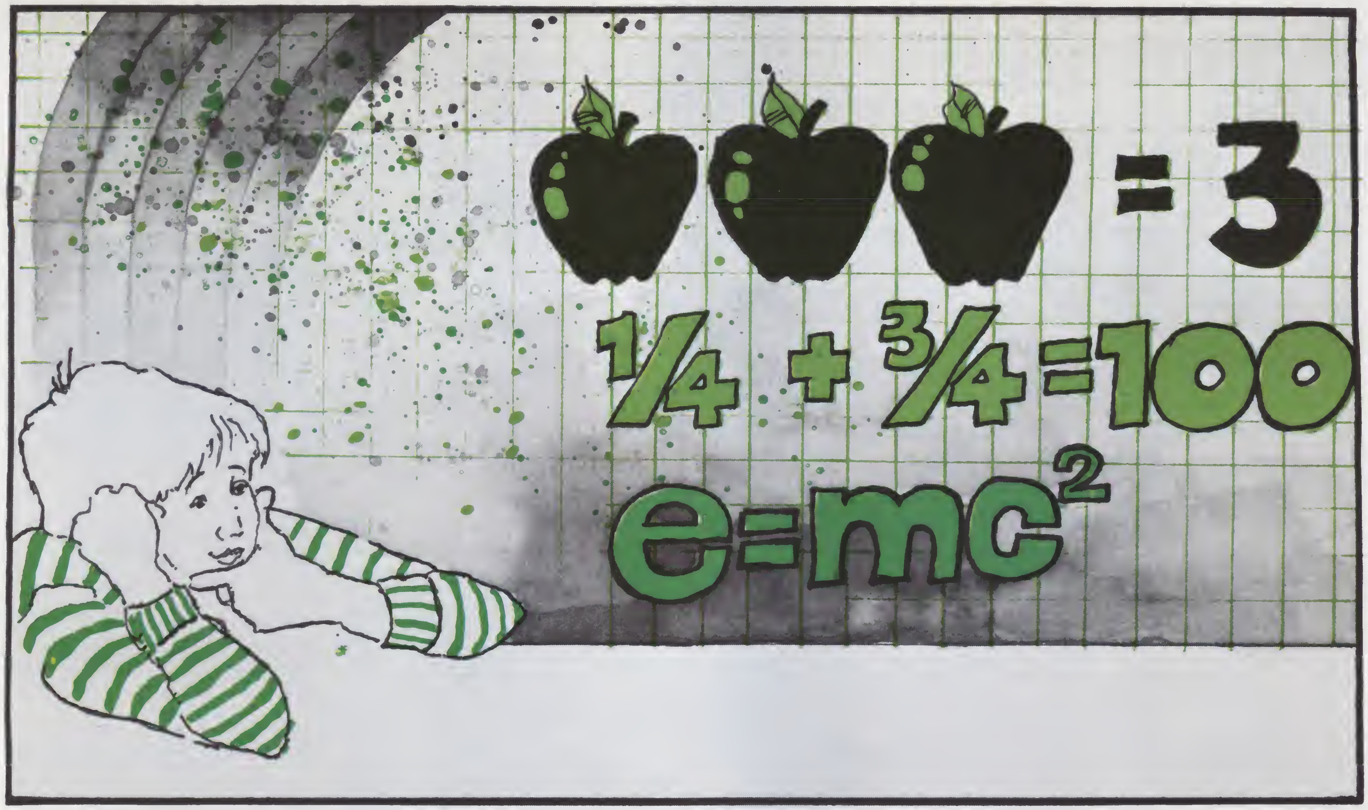
From this overview of Piaget's work, a few points should be clear. First, Piaget saw children as incapable of particular types of thought prior to particular—though not necessarily precise—chronological ages. Second, to avoid faulty internal organization, children must assimilate many experiences before being confronted with experiences needing accommodation. This implies that much needs to be organized extensively and accurately enough for new structures to be created.

A corollary to this is that we should exercise caution when selecting the kinds and sequences of experiences children will be presented with in order to help them develop the framework to handle more complex types of thinking.

Some Major Differences

Although Papert asserts that much of his work follows Piaget's precepts,

Joseph F. Rousseau, PhD, and Stephen M. Smith, EdD, are assistant professors of education at Keene State College, Keene, NH 03431.



there are some major departures.

For example, Papert implies that children can deal with increasingly abstract ideas at ages earlier than Piaget said they could. But research in Great Britain during the late 60s showed that children with some mental abilities beyond those typical for their chronological ages could not handle the underlying mental operations necessary to perform advanced tasks. These children showed no evidence of being able to do more complex mental operations until they reached the chronological *age spans specified* by Piaget (see Reference 3). This seems to be true for all children, showing that chronological age is linked to the ability to perform specific mental operations.

In another departure from Piaget, Papert says that computers will restructure learning and thinking. But Piaget would say that learning and thinking progress according to inexorable laws related to the development of new cognitive structures, which are based on the organization of action, reaction and interaction with the environment. The environment becomes much larger and more complex as a child gets older, and this complexity seems to roughly correspond—at least in American society—to Piaget's age levels.

Educators are concerned with how to manage the learning environment, because structuring the child's experiences in that environment lets them control the levels of abstraction the child must fast. A child faced with abstractions beyond his level will fail repeatedly. Papert claims that children would experience the thrill and power of their own abilities to control what they learn, but this may be counterproductive. Children in total control of their own learning may choose only narrow areas of interest on which to focus their energies.

Granted, we're moving toward even more specialization in our society, but choosing a specialization usually occurs after a person has explored numerous areas. Perhaps Papert should give this some thought.

Liberating Thinking

One particularly heartening feature of the computer as presented in *Mindstorms* is its power to simultaneously control and liberate thinking. Experiences can be focused, thus letting children create new structures to a greater extent than would be possible by any cadre of teachers. If Piaget's age spans for particular types of thinking are as rigid as research suggests, then children could rapidly gain all kinds of experiences in school curriculum areas and, within their conceptual levels, put these experiences together in many different ways. This would make for such a broad base of internal structures that clear and smooth transitions to more complex levels would remain inevitable. More internal organizations could develop than teachers and books alone could provide.

Unfortunately, Papert gives few clues as to the role of the teacher in *Mindstorms*, or, for that matter, the role of the school. Papert sees them as vastly different from what they are now—if they continue to exist—but fails to indicate how they will be different. It would be nice to hear Papert's thoughts about these important factors in education. This is one major shortcoming in the book.

It would also be interesting to hear Papert's ideas on the societal impact. Nearly one-fourth of the American population is involved on a full-time basis with schools, all the way from preschool programs through graduate and professional schools. This figure includes students, teachers, aides, administrators, secretaries, custodians, textbook salespersons, etc. Obviously, any major changes in the schools will have corresponding impacts on both society and the economics of that society.

Other Voices

Although Papert's work is related

See List of Advertisers on page 210

HOOK 3 WINNERS

DOLPHIN

$139.00
WINNER #1

$159.00
WINNER #2

$179.00
WINNER #3

data ed

WATCH THIS SPACE FOR COLOR WINNER

By TECO

DEMAND THESE UNITS FROM YOUR DEALER!

Data-Ed
18 Bridge Street
Salem, NH 03079
603-893-2047 (collect)

210

_____ TM-12GX Monitors at $179.00 plus 2.75 shipping
_____ TM-12G Monitors at $159.00 plus 2.75 shipping
_____ TM-12 Monitors at $139.00 plus 2.75 shipping

☐ Please send OEM information about your low cost custom fit smart printers.
☐ Check or Money Order enclosed
☐ C.O.D. Orders Accepted
☐ Bill my-☐ VISA-☐ Mastercharge

VISA
master charge

Name ____________________
Address ____________________
City, State, Zip ____________________

Card No. ____________________
Exp. Date ________ Bank No. ________
Signature ____________________

to the ideas posited by Piaget, it is, in fact, more closely aligned with the work of Hilda Taba. She outlined a nine-step process in which one learns how to learn and think just as one learns anything else. She specified no time restrictions during this progression from the concrete to the abstract, and viewed it as continuous rather than as having beginning and end points for each stage. Overt activities performed by the student are seen as illustrative of covert mental processes.

Papert's ideas on debugging may parallel Taba's thoughts on covert mental operations. Producing habitual debuggers may even satisfy Papert's claims for making students their own sources of motivation. One of this main points is that students will find Logo so satisfying that they will want to continue thinking, learning and being challenged. This is entirely consistent with Taba's ideas.

Papert's theories seem to also follow the synectics model of teaching and learning. This model was developed at about the same time that Papert was formulating his first ideas for *Mindstorms* 15 years ago, and combines intellectual and creative development. Synectics is a process-oriented, rather than content-oriented, theory beginning to enjoy popularity among educators. Concepts and creativity are enhanced primarily due to the learner's increasing ability to use various types of metaphors. These include personal analogies, direct analogies and compressed conflicts, usually generated by the student himself in relation to a content area specified by the teacher. This may be analogous to Papert's examples of gears and having children walk out the outlines of a house to be drawn by the turtle. If creativity is both a cause and a result of content learning, then Papert's notion of power and freedom as developed and expressed in and by children via the computer would also hold true.

Although Papert was formulating his basic ideas for *Mindstorms* years ago, he didn't seem to use very much parallel research as support for his contentions. Research findings available at that time offered some interesting notions that perhaps should have been incorporated or elaborated upon.

For example, J. I. Goodlad found that a substantial shift had occurred in the writing of curricula from general educators and administrators preparing them to content-area specialists doing this. This finding represented a shift in philosophy from what had been the case for the previous 40 years, and seemed to have as its base three major goals: (1) teaching the basic conceptual structure of each content area, (2) having students approach the content as a specialist would, and (3) introducing main concepts in the curriculum as early as possible in ways students could handle.

This would seem to provide good support for Papert's ideas, but he never mentioned this work. Even though N. H. Mackworth offered some good arguments for developing problem-finders rather than problem-solvers for computerized society of the future, Papert made no mention of this work either.

This points out a significant lack of research-backed support for the ideas presented in the book. Perhaps *Mindstorms* is based on educational research findings, but the reader doesn't have any way of knowing for sure. This is a major weakness of the book, despite Papert's descriptions of how Logo seems to work with groups who have attempted and apparently succeeded using Logo and the turtle. Hopefully, future research based on larger numbers of students in more diverse settings will support Papert's hypotheses.

What is also bothersome—and also not Papert's problem—is the propensity of educators for jumping onto bandwagons and falling for fads. A question that probably won't be answered for quite a while is whether what Papert is suggesting is going to be merely a fad, or a legitimate wave of the future.■

References

Goodlad, J. I., *School Curriculum Reform in the United States*, Fund for the Advancement of Education, 1964.

Mackworth, N. H., "Originality," *American Psychologist* 20:51–66; 1965.

Shields, J., *Cognitive Psychology*, ed. A. Summerfield, British Medical Bulletin, vol. 27, no. 3, TP 191-290, London: The British Council, 1971.

This editor, author and educator urges a cautious approach to the use of computers in education.

David Moursund: Educating the Educators

By Tom Hager

Any movement needs brain and brawn—both inventive theoretical thinking and applications-minded activism—if it is to succeed. This includes the long-awaited move to classroom computers.

But from the beginning, the field has been top-heavy with theorists. Suppe's Plato Project, Dwyer's Soloworks and Papert's Logo have all added immeasurably to a fund of ideas about how computers might be used in the classroom, but have had only a limited impact on a limited number of people in the real world. For the last 20 years talk of grand plans and great expectations has far outweighed practical application.

David Moursund does more than talk.

During the last 15 years Moursund has been quietly laying the groundwork that will help make computer-based educational changes a reality. His goals are simple: *Teach school teachers and administrators what computers are, and how best to use them.* His activities cover the board. He is the author of more than a half-dozen books and pamphlets aimed at raising the computer consciousness of the people who run our schools, an instructor, the organizer of a professional society for computers in education and the editor of *The Computing Teacher*.

"Our educational system is completely dependent, in essence, on how the teachers view the world," he says. "If we can't take today's teachers and have them adjust to using computers, to this change in the world, then it's going to be very, very hard to have the kids adjust to it."

The University of Oregon, where Moursund is a professor in the Computer and Information Science Department, has become a national center for teacher education, a mecca for instructors willing to learn how computers might change their careers.

Many who come to his sessions find that the changes are bigger than they expected. Rather than simply going over the basics—what CAI (computer-assisted instruction) is or what microcomputers cost—Moursund tells them their entire approach to teaching must change.

To explain this, Moursund has written pamphlets for school instructors and administrators in which he illustrates the basic impact computers will have on curriculum by dividing the process of problem-solving into four steps:

1. Understand the problem.
2. Figure out and represent a plan of attack.
3. Carry out the plan.
4. Understand the meaning of the results and check to make sure they make sense.

Step three is where he believes machines will have the most impact. Schools now spend too much time and money teaching ways to solve problems that can more easily be solved by calculators and computers. This means that a basic change in curriculum is necessary to reduce the development of by-hand paper and pencil skills. Instead, teachers need to stress steps one through four, to increase their emphasis on thinking and understanding.

For math teachers, he uses the analogy of students learning the art of long division. Understanding the concept of dividing numbers and recognizing the situations where division is applicable is a relatively simple matter, says Moursund, but it takes an excessive amount of time—up to two years in some cases—to teach the student how to go through the computational process, learning and relearning the paper and pencil algorithm.

"How long does it take to learn to push the button?" Moursund asks. "We can now teach people to push the button and save months which could be used to teach other things—like problem-solving and understanding."

It's a matter of *choosing the most* effective tool: your fingers, an abacus, paper and pencil or a calculator.

And the change is long overdue. The high school math classes Moursund sees his teenage daughter taking appall him.

"The totality of mathematical knowledge over the last 25 years has gone up by at least a factor of five, but the average high school student who's graduating now is learning the same materials exactly, and at the same level—maybe even a slightly lower level, actually," he says. "We have the machines now to be able to change that. Let's not waste time because there isn't enough time to learn all the things that can be learned anyway."

A Family of Mathematicians

His concern with math is quite natural. He was born into a family of mathematicians. Both of his parents taught college-level math, and his father headed the math department at the University of Oregon while Moursund was growing up.

"My initial way of looking at the

Tom Hager is a science-writing intern at the National Cancer Institute in Washington, DC.

world was purely mathematical," he remembers. "I was raised to be a mathematician, and clearly I followed in the footsteps."

Those footsteps led him to a BS in math from UO, followed by a PhD in numerical analysis from the University of Wisconsin. During all his formal academic training he never took a course in computers, but began using them on his own to help with his doctoral work.

After finishing at Wisconsin in 1963, he took a faculty position at Michigan State. He didn't teach during his second summer there, so to help relieve the tedium he volunteered to create a computer-education course for math teachers attending another professor's summer institute.

He found he loved it. "The feedback and rewards from teaching teachers and seeing that they would then take this knowledge and use it was much more satisfying than proving some esoteric theorem and getting it published in some journal that maybe two dozen people read," he says. The next summer Moursund started his own summer institute.

He brought this program with him when he returned to Oregon as a faculty member in 1967. The late 60s was a vibrant period on the Eugene campus, a time of social unrest and calls for relevance in education. Student demands that faculty members come down from their towers and get involved in the moral questions of real life struck a responsive chord in Moursund. This, coupled with his growing belief in the possibilities for computers in education, solidified his distaste for abstruse mathematical games. Theories were no longer enough. He wanted to show people what computers could really do. As a result, his dalliance with summer institutes was transformed into a major focus of his career.

It wasn't long before he started to see concrete results. In 1969 he was made head of the UO Department of Computer Science, where he instituted a master's program in teaching computer science—only the second such degree offered in the nation. By 1971 interest in the field grew to the extent that a professional society, the Oregon Council for Computers in Education (OCCE), was formed with Moursund's help. The next year he was named computer component director for a five year National Science Foundation (NSF) mathematics education project.

Computing teacher David Moursund.

But he was discovering that math wasn't the only way to use computers in the classroom. Arthur Luehrmann's theories on students designing their own software for problem-solving impressed him, as did Tom Dwyer's Solo project. Computers, it seemed, could be used in almost every academic discipline. The field was opening up, rich with ideas and enthusiasm, growing in importance. Moursund grew with it.

To disseminate the skyrocketing amount of information available, he added the title of magazine editor to his list of credentials. In 1974 he started *The Oregon Computing Teacher* as a professional journal for the OCCE.

But by 1979 his workload had grown so much that he knew the magazine either had to grow or go under. Luckily for Moursund, a small computer magazine with national circulation died at about that time, and he availed himself of the orphaned advertisers and subscribers. Oregon was dropped from the title and *The Computing Teacher* was born. Circulation jumped from 500 to 2800.

The Computing Teacher

Recent issues of *The Computing Teacher* illustrate the current range of Moursund's interests. Articles cover everything from computers in art education to surveys of state needs for computer-assisted instruction, new computer games to ways of introducing micros to grade schoolers. Regular departments in the magazine review software packages, computer literacy films and books, as well as the use of computers in the arts and humanities.

Yes, the computer is invading such formerly sacrosanct academic areas as, say, music. Here Moursund believes the machine can open up potentials that were formerly obtainable only after years of discouraging study.

"It might be that all grade school kids are able to compose music," he says. "We don't know and we don't have any way of finding out because kids that young can't *perform* music. They don't have the muscle skills yet. We may want to teach grade school kids to compose music and use a machine as the feedback and performance mechanism, so now the machine performs the music for them. This would change the way people view and participate in music. And the whole nature of music as part of one's world and education might well change because of that."

Even something as well-known and relatively mundane as computer-assisted instruction carries long-range implications for the teacher. Certainly having the student perform rote drills with a computer takes some of the load off the instructor. But there's more.

"CAI provides for a much higher-quality individual feedback," says Moursund. "It may change the balance. Instead of the teacher being responsible for what the kid learns, the kid becomes responsible for himself. If the center of everything becomes the student, then education is marvelously changed."

Computers transfigure the teacher's role from a simple drill instructor to an overall facilitator and coordinator of studies, allowing him or her to stress the human part of education—to become less of a machine.

"The human element is at the heart of education and we can't mass-produce knowledgeable people."

But most school administrators don't understand where computers can take them. Bringing microcomputers into the classroom without realizing their full potential is "a fairly standard problem," says Moursund. The present interest seems to be primarily limited to saving money and time—and certainly computers do that.

Seymour Papert estimates that students could learn 10 percent faster with computers. But, says Moursund, "10 percent is just a drop in the bucket. We could easily expect some combination of 10 percent faster and 10 percent of the stuff they don't need to learn because they have the machine available. Well, what's 10 percent of the cost of educating a child right now? That cost could provide a child with a computer to use all the time."

While schools may now be underusing computers, he also sees a danger from overuse—a potential that arises when he talks about the theories of Papert. While Moursund supports Papert's Logo system for creating a problem-rich environment in now-sterile elementary schools as "a tremendously powerful idea," he fears the Logo concept could be taken to extremes.

"Where I have some trouble with Papert is that to try and sell his idea he loses another part of himself. If you've got a particular product or project where you've put a large part of your life into it, and you're trying to convince other people it's a good idea, you tend to go overboard in one direction. Especially if people only see part of what you're trying to do. If you interpret Seymour lightly, you might think he's advocating the removal of people-to-people interaction and the development of human values. But it seems quite clear to me that he isn't."

Moursund's work with teachers has reinforced his view that humans are the key to educating other humans, aided—not replaced—by computers. "Some people think we can solve the problems of education by mass-producing machines," he says, "but I think we solve hardly any problems this way. The human element is at the heart of education and we can't mass-produce knowledgeable people."

He sympathizes with teachers who are, understandably, a little wary of accepting these new competitors into their classrooms. There is considerable resistance even to something as simple as allowing calculators into an elementary school math class. "I've just told these people that a $10 calculator is going to wipe out half of what they do in the math education part of their career. Boy, is that a threatening thing. That's just attacking their basic being," says Moursund.

That's the attitude he's trying to educate out of the educators. But not all of them are negative—many are fascinated with the machines themselves.

"I've seen nothing else in any other aspect of education that's turned on people nearly as much as computing has," he says. "That's a very strong force we've got going there."

After 15 years, the task is just beginning. The whole idea of how computers will change what we want people to learn as well as how they learn it is still relatively new.

"Our educational system doesn't understand this at all yet," he says. "That is probably what I'll spend almost all the rest of my professional life working on—trying to get this idea built into our schools." ■

ENGINEERING AND PRODUCTION CONTROL SYSTEMS

Company founder and president, Walter L. Myers (right), pictured with production engineer Joe Zellers who developed the production control software for the MSI computer system.

THE COMPANY:
MYERS SPRING COMPANY, INC. LOGANSPORT, INDIANA

Myers Spring Company, Inc. was founded 35 years ago by Walter L. Myers for the manufacture of small mechanical springs which are used widely in mechanical appliances, electrical equipment, and by the automotive, construction, and many other industries. The Myers Spring company has grown to several million dollars in annual sales and employs approximately 50 people in its production facility.

Production engineer Joe Zellers comments, "we began looking at computer systems approximately ten years ago in order to keep up with the increasing demand of order processing, custom mechanical spring design engineering, and production control. In 1975, we selected the MSI system because they were the first company in the microcomputer industry to offer the necessary peripherals which would convert a microcomputer system into a usable business system. Since 1975, the MSI system has been expanded to accommodate four users simultaneously, performing a variety of plant monitoring functions and management programs."

"The system is equipped with 10 megabytes of hard disk storage presently and we plan to add an additional 10 megabytes of storage soon. The maximum downtime has been only one or two days in the 5 years that the system has been in operation. MSI has provided excellent technical support and willingness to help us with our special requirements. We have nothing but praise for MSI, they have provided excellent system support."

SPRING ENGINEERING AND DESIGN

"All of the production at Myers Spring is performed to exact customer specifications rather than to the manufacture of standard spring products. This causes an ever increasing demand for quick and efficient design and engineering capabilities. Many parameters have to be taken into consideration in the design of any particular spring, including wire size, wire type, material modulus, spring diameter, number of turns per inch, free length, spring loading, rate, solid height, working stress, working temperature, number of operating cycles, hysteresis, resonant frequency, expansion, and whether the spring has to be ground or not. It used to take over an hour for an engineer to design a spring taking into account all of these parameters. However, with the engineering software which we have developed for the MSI system, spring design can be completed in less than one minute by simply keying in the desired parameters. The MSI computer system not only designs the spring for us but prepares a complete quotation for the customer after consideration of the material to be used, the amount of waste, which equipment the production will use, the speed of the machines, the necessary labor rate, as well as the desired percentage of profit."

The production facility at Myers Spring Co. is equipped with many automated machines for mechanical spring production.

One of four workstations where design engineering, checking of sales order status, and production control monitoring is performed.

SALES QUOTATION SYSTEM

Following the computer spring design procedure, with automatic quotation feature, the actual production begins. Each quotation is reviewed and compared to actual job cost reports on the production run in order to make any necessary refinements in the quotation system software. This feature of our system has greatly improved our ability to prepare accurate quotations and to insure profitability of the company.

PRODUCTION CONTROL/JOB COST ACCOUNTING

Each production work order is tracked by the computer system at each stage of the production process. First, each order is checked against the customer quotation for accuracy. As each order is processed, exact shop labor time is recorded, for each production machine used, and each stage of the production process. Summary reports are produced showing the total amount of material used, time used on each production machine, amount of material used, and a total cost figure for each work order.

SALES SUMMARY REPORTS

The system is designed to produce monthly sales summaries which show the amount of products sold by each salesman, complete with dates, order numbers, type of product, quantity, type of material, material cost, sales commissions, etc. Totals for each desired category and for each salesman are reproduced.

ACCOUNTS RECEIVABLE SYSTEM

Each morning, invoices are generated for orders which will be shipped that day. The accounts receivable system maintains accounts for over 500 active customers. The system produces monthly statements complete with aging of open invoices.

MULTI USER CAPABILITY

The MSI system is equipped with four user terminals presently which are available for use simultaneously by the following departments: Order department, for entering new orders and checking order status. Inventory department, used for checking to see whether a particular product has been produced previously. Invoicing/Cost Accounting, used for preparation of invoices and for entry of labor and material cost accounting information. Design Engineering, used by company engineers to design new products.

Order entry, invoicing, monthly statements, and other management reports are carried out at this workstation at Myers Spring Co.

The MSI system at Myers Spring Company is equipped with 10 megabytes of hard disk memory, dual floppy disk drives, a high speed printer, and four user CRT terminal workstations.

GENERAL LEDGER TIE-INS

The MSI system automatically prepares journals for cost accounting information and sales data which can then be posted to the general ledger. Complete income statements and balance sheets are produced by the general ledger programs on the system.

MULTIPLE MANAGEMENT REPORTS

The MSI system is used in many different areas of the company in order to provide more efficient and effective management of our production facilities. Several of the reports which we obtain from the system are: **production schedules, due list for orders, new orders list, production summary by department, salesman's reports, individual customer reports and order histories, time studies, sales quotations, design engineering, sales summaries, customer statements, general ledger balance sheet and income statements.**

Consider MSI, evaluate us, talk to our users, and we think you'll buy your business system from us!

If you would like to know how an MSI business system can help you make your business more profitable, call or write, **MIDWEST SCIENTIFIC INSTRUMENTS, INC., 220 W. Cedar, Olathe, KS 66061, 800-255-6638, Telex 42525 or 437049**

✓144

MSI ...... Helping to make your business run better.

Is the computer a threatening electronic monster or education's saviour?

Through a New Looking Glass

By Henry F. Olds, Jr.

I recently visited a fifth grade classroom in a public elementary school near Boston. I saw a microcomputer sitting on a desk, and asked the teacher how she was using it in her teaching. She said she'd only had it for two weeks, and hadn't used it at all yet.

Reacting to my obvious disbelief, she added that she hadn't been told what to do with the computer. And since she was only in the third week of her in-service course on BASIC, she was still unable to teach anything about the micro to her students.

She was a bit surprised when I suggested that some of her students might already know a fair amount about the computer, and, with a little help from the manual, could probably learn how to use it in no time at all. Boldly, I asked if she would let me check out my hunch.

She actually seemed pleased to have me there for the first tryout. I think she expected that there would be dire consequences of one kind or another if children were to use the computer in anything but some narrowly prescribed way.

Well, it turned out that some kids did know a little about how to use the computer, and could use the manual to answer their questions just like any reasonable person. And they were good at teaching each other—patient, understanding and resourceful at explaining difficult ideas.

In the hour I was in her classroom, the teacher began to see her students and the computer in a new way. I never made another visit, but I suspect that the computer has become an integral and significant part of learning in that classroom.

Two Reactions

If I mention the microcomputer to teachers these days, I am likely to receive two very different reactions. For some, the microcomputer is a strange electronic monster that threatens whatever humaneness remains in education. Because it is efficient, relatively inexpensive and easily controlled (computers cannot yet negotiate for wage increases), this group is convinced that school systems will soon replace human teachers with computer teachers. They see the computer as the latest (and surely the most formidable) attempt to standardize, homogenize, trivialize and control education.

The teachers in the other group feel that the microcomputer in education is one of the most exciting developments in their professional careers. They think that this technology can help them teach more fully and effectively. They are not blind to dangers of misuse of the computer, and are willing to work to avert this. They see it as a means to liberate curriculum from the stultifying routine of textbooks and worksheets.

Working with computer programs, orchestrating the increasingly wide range of computer software, designing (and sometimes developing) new software, sharing software with other teachers—all such activities are giving these teachers a renewed sense of the professional contribution they can make through their own creativity and efforts. At a time in American education when there is substantial public criticism, declining public confidence, reduced funding, substantial reductions in teaching staffs in many communities and widespread talk of "teacher burnout," the microcomputer has been welcomed by these teachers as a fresh breath of air.

What leads to these two vastly different points of view? Mostly it's experience, and, to some degree, the quality of that experience. Microcomputers have not been around long, so only a small group has had the chance to have exposure to them. Because of the microcomputer's abilities to manipulate numbers and carry out calculations at high speeds, math and science teachers have been among the first to explore microcomputer use. This, combined with the enthusiasm of devotees that often approaches religious fervor, has tended to put off many who see themselves as humanists.

Then there is the image of the computer as a vehicle for playing video games. Many parents and teachers cringe at the thought of another device that keeps children from doing anything "thoughtful, creative or educational." The influence television

Address correspondence to Henry F. Olds, Jr., Cognitive Research Group, Education Development Center, Newton, MA 02146.

has on their children is already a concern. They would like to keep the number of cathode-ray tubes to a minimum.

Add to this a phenomenon that accompanies new developments in our society—the creation of a new vocabulary, or jargon. Experts often seem to delight in murky communication to keep their newly discovered knowledge protected from the common person.

A Humanist's Approach

For most people the world of computers is not yet easy to enter. It wasn't for me. I am a humanist. I used to teach English and creative writing. I did my graduate work in psycholinguistics. I once helped to develop a language arts curriculum. I have designed classroom environments. For the past ten years, I have concentrated my work on the in-service training of teachers. Why would I be interested in computers in education?

My initial interest was aroused because I began to realize through talking with professional associates and reading in professional journals that computers were quickly becoming a part of the educational scene. I sensed that their presence would have some major implications for staff development. The emergence of the microcomputer was, it appeared, a significant challenge to the status quo. It seemed to me that I could help teachers meet this challenge only if I understood something about the new technology.

Initially I was very skeptical. I see teaching as a fundamentally human interactive process. Supporting the process are books, which do a fine job of conveying information. What could computers add to justify their relatively high cost?

In sampling computer programs designed to instruct, I could often easily convince myself that the technology added nothing worthwhile to education. I was at times horrified by what seemed to be highly inappropriate applications.

I might well have given up on the computer had it not been for a special and haunting quality of the technology that kept me fascinated. At first I couldn't put my finger on it. But gradually, discussing uses of the computer with colleagues, teachers and friends, I began to realize its true educational significance. Most educators naturally assume that the educational function of the computer is to instruct a person in some content. The person learns what the machine teaches. What I came to realize is that this teaching function of the computer is the least interesting. It functions best in a totally opposite role—as an educational tool. *The computer does the learning and the person tries to teach it something!*

This was what I was doing when I was struggling to learn BASIC. I was learning a language that would allow me to teach the computer how to do what I wanted it to do.

It wasn't easy. But what tremendous excitement I felt when my first reasonably complex program actually worked. I was forced to understand thoroughly what I thought I knew. Correcting the bugs in my program forced me to learn even more about my topic and about my way of thinking about it. By the time I was finished, I knew the topic better than I had ever thought I could. I had learned by teaching the computer.

It was very much like what I once learned about teaching students: I had to relearn what I thought I knew to teach it well. Except I found the computer a far harder task-master. It is truly a dumb machine. It would only do what I learned to instruct it to do, and nothing on its own. It would take everything I said literally. It would persistently and painfully reveal to me every single small lapse in my thinking—always a frighteningly accurate reflection of the workings of my own mind.

At about this stage in my learning, I became curious about teachers' reactions to microcomputers. What had those with experience learned, and

how did they view the future use of this technology? What would the reaction of the inexperienced be to their first encounter with computers, with educational software and with knowledgeable peers?

My colleagues and I designed a small study to explore these questions. (See References.) We asked 18 school teachers from a variety of backgrounds and with a wide range of computer experience to give us their reactions to and reflections on half a dozen different instructional applications.

We chose our programs carefully; we wanted them to represent the broadest possible continuum of types that could be considered, and we wanted to represent clearly six types along that continuum. We were interested to see if teachers would perceive the same continuum that guided our program selection.

The continuum we had in our minds grew out of a theoretical framework we were building to structure our own thinking about instructional uses of the computer.

This framework also provided us with a convenient language to talk about computer programs. One of the things we found in our discussions with the teachers was that the framework did fit and helped to clarify their experiences as well as ours. Since that first study, we have used this framework many times to help people understand the richness and diversity of instructional uses of computers.

This framework starts with known and relatively familiar uses and then moves to unfamiliar and largely unexplored uses. It is based on a range of experiences with computer programs that will not yet be widely shared. I could discuss more easily the varieties of educational television because I could count on a wealth of shared experiences. In this case, I will have to ask the reader to trust a bit in my experience and in the experiences of those teachers with whom I worked. Hopefully, the reader will soon have a chance to check out our perceptions against his or her own experiences with computers.

Theoretical Framework: The Educational Uses of Computers

Many, if not all, present educational uses of computers fall into one or the other of two categories: the computer as a medium or the computer as a tool. The former means using the computer to convey to the user, or to instruct the user in, some body of knowledge. The latter involves using the computer to accomplish some task for the user, including the most significant task of creating new tools.

There are three broad categories of the use of computers as an instructional medium: tutorials, games and simulations.

Tutorials—There is a rather long history of attempts to use the computer to instruct students in a direct and explicit way (e.g., computer assisted instruction). Implicit in CAI is the assumption that the program can be a good teacher. Tutorials using multiple-choice answer formats generally assume that there is one correct answer to every question, and pay little attention to what may have led a student to a wrong response.

On the other hand, tutorials which attempt to analyze and evaluate a student's constructed answer must do so with a theory of the nature of knowledge in the domain being taught and a cognitive-developmental theory of the learner. Rarely are these theories well-developed or explicit.

Furthermore, it is obvious to any teacher that even good tutorials are merely effective at mimicking a small number of attributes found in good teaching.

Those who seek to design tutorial programs assume a heavy burden. However intelligent they may try to make the programs, it is unlikely that they will be able to cope with the nuance and unpredictability of human thought, let alone its convoluted nature. Furthermore, the cost of a computer's misunderstanding of a user is too great to be tolerated in educational settings. A teacher who misunderstands a student's behavior can always make amends in light of additional information. I have not yet seen a self-correcting computer program.

The immediate dilemma, however, for teachers who are searching for good instructional programs is that most available tutorials are merely textbooks transferred to computer disks; the computer is asked to function as a high-priced page turner. And the content presented doesn't even get a chance to pass through the often-beneficial filtering process of a real teacher's intelligence. ("Students, just do the odd-numbered examples on page 34, and then skip to page 38.")

Teachers question why there should be program after program of drill and practice, or question and answer. They question why a computer is needed to carry out mundane tasks, many of which ought to be reconsidered in the first place, and not unthinkingly proliferated through a new medium. They sense that there is great irony in the use of this powerful tool for drilling students in performing an algorithm for long division when the technology itself has made the use of the algorithm all but obsolete.

Nevertheless, computer programs can be effective tutors. As a musician, I think the Music Theory tutorial programs by Linda Borry, available through the Minnesota Educational Computer Consortium (MECC), are excellent. My colleagues and I discovered that a simple tutorial on fractions could provide the user with some real insights into the nature of fractions. Fig. 1 shows a partial "run" of this program.

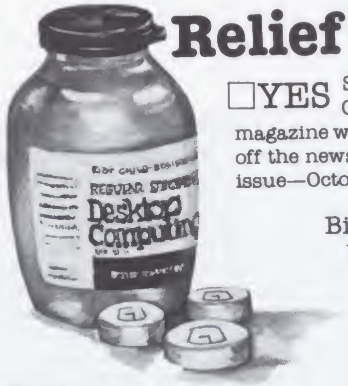
At Trial 6 most users (adult and child) stop and scratch their heads. "How can there be a fraction between 6/16 and 7/16 that has a denominator of 16 or less?" What seems to happen in many cases is that the person's search strategy, which suggests that the next approximation would be 13/32, gets in the way of finding a solution. Many people can't pass this point without *inventing* the decimal system and discovering that 4/10 (or 2/5) is possible—but, in this case, wrong. It takes a little more perseverance to find that an answer in this case is actually 6/15. Almost all persons are astounded to discover (or rediscover) that all of the following fractions lie between *6/16 and 7/16:*

4/10 (2/5)
5/12
5/13
6/14 (3/7)
6/15

The point is that it is possible to create reasonable and effective tutorial programs for computers. But so far, the focus has been almost exclusively on tutorial use, which may be at best limited and at worst, if done unintelligently and insensitively, downright dangerous.

Games—It is worth distinguishing between two categories of games: those which attempt to convey some portion of the content of some discipline (content games) and those which attempt to sharpen the use of a cognitive strategy that may be applicable to a variety of disciplines (process games).


HISTORY
REPEATS
ITSELF



Relief at last . . .

☐ YES Sign me up as a Charter Subscriber to Desktop Computing. I will get one year of the only computing magazine written in plain English for $17.97 (a savings of 50 % off the newsstand). My subscription will begin with the premiere issue—October 1981.

Bill ☐ Me ☐ MC ☐ VISA ☐ AE

Your first issue will be arriving by mid-October 1981

Canadian 1 yr. only/US funds $19.97
Foreign 1 yr. only/US funds $29.97

Name ______________________________

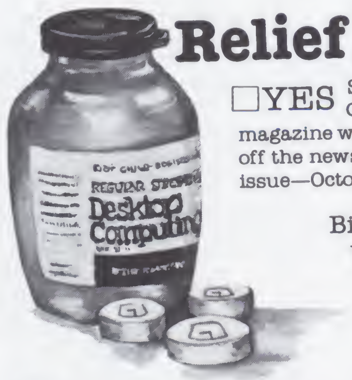
Address ______________________________

City ____________________ State __________ Zip __________

Signature ______________________________ Expire Date __________

Card # ____________________ Interbank # ____________________

DESKTOP COMPUTING Peterborough, New Hampshire 03458 319C7B



Relief at last . . .

☐ YES Sign me up as a Charter Subscriber to Desktop Computing. I will get one year of the only computing magazine written in plain English for $17.97 (a savings of 50 % off the newsstand). My subscription will begin with the premiere issue—October 1981.

Bill ☐ Me ☐ MC ☐ VISA ☐ AE

Your first issue will be arriving by mid-October 1981

Canadian 1 yr. only/US funds $19.97
Foreign 1 yr. only/US funds $29.97

Name ______________________________

Address ______________________________

City ____________________ State __________ Zip __________

Signature ______________________________ Expire Date __________

Card # ____________________ Interbank # ____________________

DESKTOP COMPUTING Peterborough, New Hampshire 03458 319C7B

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 82 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Desktop Computing™
A WAYNE GREEN PUBLICATION

80 Pine Street,
Peterborough, N.H. 03458

Attn: Debra Boudrieau

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 82 PETERBOROUGH, NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE

Desktop Computing™
A WAYNE GREEN PUBLICATION

80 Pine Street,
Peterborough, N.H. 03458

Attn: Debra Boudrieau

Content games generally appear to be more instructive because it is frequently easier to grasp what is being taught, but they often lack the playful appeal of process games. In fact, content games often turn out to be tutorials in "game" clothing. Well-designed content games can be very valuable, but usually, once the content is learned, there is little reason to continue to play.

Process games tend to be more successful and long-lived. But it is not easy in some cases to describe the process being learned. Furthermore, it has often been noted how people can play process games without learning seeming to occur at all. Mere repetition of a process does not of itself guarantee improvement in that process.

The skeptical teacher wonders why the computer is needed for playing educational games. Haven't teachers been using effective educational games for years without the computer's help? What special value does the computer now make available?

There are not now a large number of successful educational games of either content or process types that use well the computer's special abilities. They are not easily sold to an educational system that discourages the use of any games because it does not consider play to be significant learning. Perhaps the increasing availability of computers will stimulate the invention of a new generation of worthwhile educational games, and perhaps playful learning will be more acceptable when masquerading as computerized instruction.

Simulations—Many of the physical or social phenomena we want to learn about have spatial or temporal scales that make it impossible, or highly inconvenient, to study the real phenomena directly. As a way of beginning to understand such matters as planetary motion, Aztec culture or voting behavior in a coming election, we build simplified models to simulate the real systems. These simulations can then be studied to gain insight into the realities they reflect.

A well-constructed simulation can be highly instructive, and microcomputers lend themselves to the presentation of simulations. But because simulations simplify reality, they need to be considered critically and used with caution. Uncritical acceptance of any simulation (and its simplified underlying model) can create misunderstanding rather than insight. As any scientist will point out, the ultimate resolution of a discrepancy between a model of nature and nature itself must always be made in favor of nature.

Skeptical teachers worry that students who become too immersed in a world of computer simulations may get dangerously out of contact with reality. In agreement, I respond that even if we had an abundance of superbly designed educational simulations (which is far from the case), we would be well advised to help our students develop a finely tuned critical capacity well-grounded in experiences with reality.

For example, I do not know how to fly a plane, but I have recently been learning something about flying using a flight simulation program which models in very simplified form the experience of flying. I think I have learned a lot, but it would be both foolish, and possibly dangerous for me, to confuse what I know with actually knowing how to fly.

One final caution about simulations. I have frequently found that what students learn from simulations is not at all what the simulation was designed to teach. A student's fascination with a simulation, as with a process game, often results from pleasurable repetition of habitual responses (e.g., note children's fascination with arcade game simulations). There is one educational-simulation program I have worked with many times, and I still have not learned what it was designed to teach. I enjoy using it because the graphics are clever and funny. Could it be that the clever graphics are interfering with my learning?

The Computer as Tool

Special-purpose tool—There are many computer programs designed to carry out a specific task and require from the user only responses to predetermined questions. Such special-purpose programs are common in the business world. They handle such regular tasks as inventory control, accounts receivable, mailing lists and telephone directories.

Such programs, designed to help the user solve a particular type of problem, can be useful, particularly if the problem is both significant and recurrent for the user, such as keeping grade records on students. A program called Apple Gradebook is a special-purpose tool that many teachers have found useful. Another example is a program designed to provide a continuous graphic representation of temperatures over time, which is valuable for science education.

As time goes on, and as teachers and administrators can describe special-purpose tools that would be useful for various aspects of education, many more programs of this kind will be available.

General-purpose tool—There are two general-purpose, symbol-manipulating tools that the micro-miniaturization of digital electronics has made widespread. The hand calculator is everywhere, and its utility is well accepted. Even public educational institutions have grudgingly admitted its

```
                              FRACTION SEARCH

Your mission is to find a fraction between 0 and 1.
The denominator of your fraction should not be greater than 16.
      0---------------------------------------------------------------1
Trial # 1 ?   1/2
              1/2 is too big
      0-------------------------------------------------------------1/2
Trial # 2 ?   1/4
              1/4 is too small
    1/4-------------------------------------------------------------1/2
Trial # 3 ?   3/8
              3/8 is too small
    3/8-------------------------------------------------------------1/2
Trial # 4 ?   7/16
              7/16 is too big
    3/8------------------------------------------------------------7/16
Trial # 5 ?   6/16
              6/16 is too small
   6/16------------------------------------------------------------7/16
Trial # 6 ?
```

Fig. 1. Sample run of a fractions program.

value. Not common in education as yet, but quickly displacing the typewriter in many business offices, is the word processor, a general-purpose tool with equal or greater potential for influencing our lives than the hand calculator.

The microcomputer makes these general-purpose tools and a good many others (e.g., data-management programs, financial-management programs, graphics systems, authoring languages) available to the user through a single form of technology with appropriate peripheral equipment. For example, to function as a word processor, the computer must have an appropriate word-processor program, a mass storage medium and a printer.

General-purpose tools frequently amplify human capacities well beyond their obvious pragmatic advantages. The word processor provides the user with the ability to modify and rearrange text with ease. Insertions and deletions no longer require the writer to "cut and paste" scraps of paper. Beyond these advantages, every regular user of a word processor that I know has found a new measure of freedom with words. Experiments using a word processor with young children indicate that the potential impact of word processors on the education of the young is likely to be profound. A recent report from the National Institute of Education Study Committee on the projected impact of word processors and automated dictionaries shares my optimistic view of the important future roles this tool will play.

Because educators have so far been preoccupied with using the computer as a tutorial medium, few are aware of the wide range of general-purpose tools that are available and have not considered their educational potential. Nor have they considered how the general availability of such powerful tools in the near future will alter the world of the students they are educating. It is entirely possible that use of such tools will soon become a new basic skill. Therefore, when I am asked by schools what educational software they should purchase, I strongly advise them to purchase at least one well-regarded program from each of the major categories of general-purpose tools:

- Word processor
- Database manager
- General financial manager
- Graphic interpreter (for data plotting)
- Graphics creator
- Music creator
- Authoring system

I recently observed junior high school students working with a general-purpose tool that is still under development by a research project with which I am associated (the Dimensional Analysis Project, Cognitive Research Group, Education Development Center, 55 Chapel St., Newton, MA 02160).

It is called SemCalc, which stands for semantic calculator. It allows a student to use the computer to carry out calculations involving both numbers and the units to which the numbers refer. For example, consider the following typical word problem from a mathematics textbook:

Sally bought 5 boxes of crayons. Each box contained 23 crayons. How many crayons did Sally buy?

Using SemCalc, the student enters the relevant information on a com-

puter *scratchpad*. The scratchpad asks the student to enter information to answer both the questions "How Many?" and "What?" So, in this case, the student would enter, "5 boxes" and "23 crayons/box." When satisfied that there is enough information on the scratchpad to solve the problem, the student chooses the operation mode. Here he or she chooses which quantities from the scratchpad are to be operated on and what operation is to be performed. Assuming the student chooses the correct quantities and operation, the following information would be shown:

The units of your answer are:
crayons

If these units were not the ones the student needed to answer the problem, then it would be possible to do the operation again. In this case, the units are correct, and when the student confirms this, the numerical portion of the answer is added—115 crayons—and the answer is added to the scratchpad in case further calculation with the new quantity is required.

Our research project is exploring whether the availability of such a tool will help students solve (and, more important in the long run, think correctly about) all varieties of real problems that lend themselves to mathematical solution. Though our research is not complete, it is obvious that once students realize how this tool makes it possible to explore the relationships among the quantities involved in a problem, their approach to problem-solving becomes an engaging act of discovery rather than an unpleasant effort to remember and apply correctly the "right" formula.

I have been astonished and impressed by the delight students experience when the use of this tool allows them to learn that their intuitions about how to solve complex problems often turn out to be correct. I have been equally impressed by the perseverance of most students in using the tool to eventually get a correct answer even after many false starts and incorrect answers.

If we are limited to the relatively familiar uses discussed so far, there would be little reason to expect that the effects of this technology would be very different from those that followed the introduction of other technologies into education, such as the textbook, the audio recording or television—each has uses as both medium and tool. The feature of the computer that makes it different in kind is that it immensely extends each individual's capacity (as well as the capacity of any collection of individuals) to create new tools. Each of us now has, or will soon have, the opportunity to use the computer to design a tool to fit our own perception of a task we want to perform or a problem we wish to solve.

As noted earlier the computer is ours to teach—an always-available, ever-patient genie to do our bidding with no limit on the number of wishes allowed. It is not surprising that we may all have some difficulty in imagining how we can use such a tool or what it will mean for our lives. None of us has ever before had such an opportunity.

For our children, such experiences may be commonplace. For example, a generation from now, an educated person might consider a procedural approach to problem-solving natural and commonplace, be comfortable with many strategies for structuring

data and representing knowledge and regularly create unique tools for applying these strategies. Or, to put it another way, while we may be concerned about how to use the computer to teach students, students will undoubtedly be learning how to teach the computer, *with or without our help*.

The most impressive current example of this trend is seen in those young students who have had the chance to use the Logo language developed at MIT by Seymour Papert and his colleagues. These students have learned in a remarkably short time to teach the computer to make tools that they can then use to carry out totally new procedures of their own devising. As Papert so rightly points out, Logo lets these children begin to realize the most powerful idea of all—that each and every one of them is capable of powerful ideas.

Such easily accessible and usable tool-making languages are likely to be more prevalent in the future. Meanwhile, it is possible, to a degree, to use the general-purpose tools noted above as toolmakers—to add one's own ideas to the powerful ideas the tools embody and thereby create new tools.

The Software Dilemma

Everyone in education is looking for good educational software, but the sad truth is that little exists, and what is about to become available won't change this situation much. Part of the dilemma is a function of economic realities. The creation of quality software of any kind takes large amounts of time and money. Software developers are not sure the educational market warrants the investment required. Both the federal government and private foundations have far less to spend on education these days and are far more cautious about how they spend what they have.

Those few software developers who have turned their attention to education are clearly doubtful about whether teachers would appreciate quality if they could have it. They seem to believe that teachers will always opt for familiar and traditional educational materials, even if offered a choice that includes far better products.

We will soon have available software products from all the major educational publishers. From the small sampling I have so far seen, I would advise very thorough review and much thought before making an investment in this material. Much of the material might just as well be in a book. And the content errors, program bugs, authoritarian attitude toward the user and poor pedagogy reflect the haste with which this material has been rushed to market and the low regard the publishers have for teachers and children.

This "they wouldn't know a good one if they saw one" attitude has been proven false in the work I have done with teachers, including the study mentioned above. Teachers do have insight and sensitivity about appropriate uses of computers and what makes for quality software. They know as well as anyone the failures of many traditional educational curricula and methods, and they understand that better educational materials must take into account the subtlety and variety of human learning. They *do* sense the power of computers to make feasible intellectual activities heretofore unwieldy, if not impossible. They *can* recognize quality software and make good choices if those choices are available.

Educational Software—Some Quality Standards

Discussions with teachers and colleagues have led me to spell out some standards all software ought to meet. It is certainly clear that the creation of quality educational software requires extensive and sophisticated understanding of every area of the endeavor. To avoid the weaknesses of the past, knowledge of educational theory, educational practice, curriculum design and computers must be carefully integrated. And this knowledge must be blended with a sensitivity to learning and to learners.

Content—Effective instruction must be built on a well-conceived model of the discipline. A great many key ideas, once understood, are simple, elegant and powerful. They must be communicated in such a way that the student can grasp their simplicity, appreciate their elegance and feel their power.

Often, teachers point out, diverse instructional approaches are required to communicate an important concept effectively and deeply. The concept must be given many forms of expression; the student must have many opportunities to experience the concept. The computer can make available to teachers and students a greater variety of instructional forms than has ever been accessible before.

Cognitive Development—The computer allows and encourages a "constructive" approach to learning. It is not the only approach, but it may be the most powerful. It makes one fundamental assumption: the learner constructs knowledge. In this approach, the child's understanding of reality forms a base on which he or she can build intellectual tools needed to further that understanding. Learning takes place when the child has the opportunity to construct new knowledge.

Educational software should provide opportunities for such knowledge constructions to occur. It should give the student the chance to invent ideas and play with them. At the same time it can help the student develop concepts and skills that fall within traditional definitions of the disciplines. Creating such software requires both formal knowledge of research on children and learning, and informal knowledge teachers develop through their contacts with kids.

Teachers feel that good software ought to be nonjudgmental. It should not interfere with or constrain the student's instincts, nor should it insult his or her intelligence. It should have a fluidity that matches the fluidity of the mind. It should help students reach beyond conventional ideas or formulations by encouraging creative guesswork, speculation and playful trial-and-error learning. It should provide feedback, but should not pretend to be a teacher. How a program responds to a student's errors is a critical indication of how it views that student's learning.

Pedagogy—The construction of new knowledge, as Piaget has noted, is an interactive process, a dialogue carried on by the learner with both the physical and social worlds around him. Pedagogy is concerned with the nature of the interaction that takes place and seeks to create the conditions that enhance communication with the learner. Since the computer is fundamentally an "environment" for interaction, knowledge of effective pedagogy can ensure that its use supports the process of learning.

As noted, teachers are sensitive to the difference between software that is pedagogically sound and software that is not. Furthermore, they feel that software should not dictate the teacher's pedagogy but should adapt to it. Software should involve the teacher, not exclude and bypass the teacher. Not all educational materials should become software. The ideal

Now NRI takes you inside the new TRS-80 Model III microcomputer to train you at home as the new breed of computer specialist!

NRI teams up with Radio Shack advanced technology to teach you how to use, program and service state-of-the-art microcomputers...

It's no longer enough to be just a programmer or a technician. With microcomputers moving into the fabric of our lives (over 250,000 of the TRS-80™ alone have been sold), interdisciplinary skills are demanded. And NRI can prepare you with the first course of its kind, covering the complete world of the microcomputer.

Learn At Home in Your Spare Time

With NRI training, the programmer gains practical knowledge of hardware, enabling him to design simpler, more effective programs. And, with advanced programming skills, the technician can test and debug systems quickly and easily.

Only NRI gives you both kinds of training with the convenience of home study. No classroom pressures, no night school, no gasoline wasted. You learn at your convenience, at your own pace. Yet you're always backed by the NRI staff and your instructor, answering questions, giving you guidance, and available for special help if you need it.

Training includes new TRS-80 Model III microcomputer, 6-function LCD Beckman multimeter, and the NRI Discovery Lab with hundreds of tests and experiments.

(TRS-80 is a trademark of the Radio Shack division of Tandy Corp.)

You Get Your Own Computer to Learn On and Keep

NRI training is hands-on training, with practical experiments and demonstrations as the very foundation of your knowledge. You don't just program your computer, you go inside it...watch how circuits interact...interface with other systems...gain a real insight into its nature.

You also work with an advanced liquid crystal display handheld multimeter and the NRI Discovery Lab, performing over 60 separate experiments. You learn troubleshooting procedures and gain greater understanding of the information. Both microcomputer and equipment come as part of your training for you to use and keep.

Send for Free Catalog... No Salesman Will Call

Get all the details on this exciting course in NRI's free, 100-page catalog. It shows all equipment, lesson outlines, and facts on other electronics courses such as Complete Communications with CB, TV/Audio and Video, Digital Electronics, and more. Send today, no salesman will ever bother you. Keep up with the latest technology as you learn on the latest model of the world's most popular computer. If coupon has been used, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016

All career courses approved under GI bill.
☐ Check for details

172-091

NO SALESMAN WILL CALL.
Please check for one free catalog only.

- ☐ Computer Electronics including Microcomputers
- ☐ Color TV, Audio, and Video System Servicing
- ☐ Electronics Design Technology
- ☐ Digital Electronics
- ☐ Communications Electronics • FCC Licenses • Mobile CB • Aircraft • Marine
- ☐ Basic Electronics
- ☐ Small Engine Servicing
- ☐ Appliance Servicing
- ☐ Automotive Servicing
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Heating, Refrigeration, & Solar Technology
- ☐ Building Construction

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council

relationship between software and print material should be carefully considered.

Technology—To take advantage of the possibilities of microcomputer technology requires a thorough understanding of the range of uses to which this technology can be put, both as medium and as tool. It also requires recognizing that this technology is significantly different from all former technological advances that have had so little impact on education.

We are probably still in the Model T stage in the development of this technology. Teachers hope that those creating software have some knowledge and vision of how the technology will be changing. They sense that software designed within the constraints of existing hardware may quickly become obsolete. But powerful, general-purpose software that can be adapted to evolving capabilities is more likely to endure.

Software Design—The nature of the computer is that it is interactive and therefore educational. Good educational software will grow easily out of an understanding of this nature. When the design of software is in conflict with the nature of the computer (as is the case when print materials are turned into software), the product reflects the tensions of the forced marriage.

We must think about curriculum and learning in ways that are compatible with a technology that lends itself to our efforts. We must not simply overlay our ideas of curriculum and learning on the new technology.

Teachers emphasized several aspects of good design. Software should have a clear purpose, which includes why the computer is appropriate for the task. It should, wherever possible, use all available modes to present content: text, graphics, auditory and kinesthetic. It should allow a student to review his or her recent work. It should provide the student with help at any time (the ultimate referral should, of course, be to the teacher). It should take advantage of dramatic possibilities that interaction with a computer can afford ("I wonder what will happen next?"). And, it should not be limited to one-on-one interactions. Software can be designed to engage two or more students in a collaborative learning endeavor. Such shared learning has many benefits (not the least of which is proving that computers are not necessarily antisocial).

Focus on In-Service

The personal microcomputer (what one writer has appropriately called the "mass market micro") is relatively unfamiliar to most teachers. How it and its uses are perceived depends on how these are introduced to teachers. In other words, the nature of in-service training will be a most significant factor in how computers become part of education.

Most of my professional career in education has been devoted to designing and putting into practice improved staff development programs. I believe that the quality of education a teacher receives, whether preservice or in-service, affects the way he or she teaches. What a teacher experiences in training is frequently transferred into practice.

So, if a teacher comes to know computers by taking a course in BASIC, chances are the teacher will introduce students to computers by teaching them BASIC. Furthermore, since

those who teach computing to teachers frequently believe that their way is the best or only way, the teachers they train will likely believe that learning BASIC is the only way.

I have thought a lot about how best to provide teachers with good in-service training in computers. There is only one thought about which I am absolutely certain: *there is no one best way for everyone*. I also agree with Jim Edlin, one of the more perceptive writers on microcomputers: "In microcomputing lies the seed of an invention with the power to catalyze major beneficial changes in our lives; this invention doesn't stand a chance of reaching fruition until it is wrested from the hands of 'computer people.' " I would add that it stands its best chance of reaching fruition if put in the hands of teachers—teachers should be deeply involved in designing and providing training programs.

A Few Suggestions

A few suggestions have been made that I think are decidedly *not* the answer for everyone. They may be of some worth at some point to some people. But they have real weaknesses as material for building a strong foundation for understanding computers.

One suggestion is that we should be teaching teachers (and, by extension, students) something called "computer literacy." Such literacy is frequently seen as consisting of a body of knowledge about the history of the computer, the nature of computers and their components, the languages of computers, the impact of computers on our lives, the ethical and legal issues raised by computers, etc. Courses in computer literacy are proliferating, and teachers and students are learning a lot *about* computers, which is very different from learning how to do something *with* computers.

Learning information about computers is a typically narrow academic view of literacy. It fails to comprehend the impact that computers are having on our lives. Most of us achieve a reasonable level of functional literacy in technological domains that affect our daily lives without taking "literacy" courses. The literacy associated with automobiles, television sets, high-fidelity systems, microwave ovens, typewriters and calculators grows out of experience with these tools. So it will surely be with the microcomputer. In this sense, many young people are already more computer literate than their elders because their experience with computer applications (through toys, television shows and arcade games) is more diverse.

There are *deeper levels of understanding* computers—the "computer sciences." These are fine for training computer scientists. But if a diluted general computer science is recommended as a means of achieving computer literacy for teachers, great care must be taken not to make the same mistake that plagued the teaching of science for years; that is, confusing real understanding with the capacity to name things. As David Hawkins, an educator and philosopher of science, has pointed out, when a child asks a question about some phenomenon and is immediately given a name for an answer, a knot is tied in his head that shuts off inquiry and prevents understanding. Computer science that ties people's heads in knots with terminology and shuts off real inquiry into the nature and function of this new technology must be

avoided at all cost.

There are others who suggest that we ought to be teaching teachers and students how to program the computer in some high-level language (BASIC being the strongest contender). For example, Harvard University has recently made it a requirement for graduation that all students demonstrate the ability to write a short computer program that will function properly on a computer (they have some choice of languages). I have no doubt that teaching some form of programming in a high-level language has great merit in beginning to build an understanding of what makes a computer function. For some, depending on the particular language chosen, it is an excellent way to begin.

But, as suggested above, it is not at all obvious that a student needs a course in BASIC or FORTRAN or APL. The programming language called Logo gives very young students the opportunity to write programs within a few minutes of sitting down at the computer. My daughter recently learned some of the fundamentals of programming by playing with a commercially available toy called Big Track by Milton Bradley. Marilyn Burns points out that there is much about the nature of programming that can be learned without having a computer at all.

If we focus too intently on teaching teachers or students any particular high-level computer language like BASIC, we also run a few risks. Some computer experts predict that most languages will be replaced by other languages that can be more easily used and will be more effective in carrying out diverse functions.

Second, since it is unlikely that most people will wish to spend the time that it takes to write computer progams for moderately complex tasks, too much focus on the learning of higher-level languages could be distracting and confusing. Instead, time might be well spent becoming adept at using the general-purpose tool "functional languages" (sets of commands) required to use the computer as a word processor, a database manager or a financial planner.

Some argue that by learning one of the relatively simple authoring languages (e.g., Pilot or Tutor), one can easily create programs to teach large parts of the existing curriculum or to provide drill and practice after a topic has been taught.

Such an approach can be taken, and it may have some value. But the question must be asked: Why do we wish to limit this powerful technology to doing things that were done without such expensive and complicated hardware? Hopefully, it is only for lack of knowing what else to do that so much attention is given to using the computer in this way. Discussions above of the variety of ways of using the computer may put this suggestion in proper perspective.

Finally, there are a few voices—more and more as experience grows and understanding increases—who suggest that we should devote most of our energies to understanding how computers can (and undoubtedly will, whatever our intentions) expand the intellectual capabilities of learners. This role of the computer, as an intellectual amplifier, is promising. It requires the recognition that this technology is more significant to growth of human consciousness than any other development experienced by anyone alive today. New technologies change the character of human competence and culture, as McLuhan and others have observed, not merely because they add on new ways of doing old things, but especially because they make possible the doing of new things. *That computers can make us creators of our own intellectual tools is difficult to express and understand, but it is the central idea for educational progress.*

As I proposed at the beginning, teaching the computer provides a way of turning a mirror on our own thinking. If teachers can understand and benefit from the reflections they see, they may begin to appreciate the power this technology offers to education. And, most important, they may allow their students to explore the potential of computers in much the same way.■

Some computer experts predict that most languages will be replaced by other languages that can be more easily used and will be more effective.

References

Henry F. Olds, Jr., Judah L. Schwartz and Nancy A. Willie, "People and Computers: Who Teaches Whom?" Education Development Center, Inc., September, 1980. (Note: Copies of the report are available from Cognitive Research Group, EDC, 55 Chapel St., Newton, MA 02160. Send $3.25 to cover cost of printing and postage.) Support for the research was provided by Control Data Corporation, Minneapolis, MN.

Apple Gradebook, Creative Computing Software, Morris Plains, NJ 07950 (32K Applesoft, Disk, $24.95).

Robert Tinker, Technological Educational Resource Centers (TERC), 8 Eliot St., Cambridge, MA 02138.

Automated Dictionaries & Word Processors. The report of an NIE study committee chaired by George A. Miller, NIE, Washington, DC, 1980.

Daniel H. Watt, "A comparison of the problem-solving styles of two students learning LOGO: A computer language for children," *Creative Computing*, Dec., 1979, p. 86.

Sylvia Weir and Daniel Watt, "LOGO: A Computer Environment for Learning-Disabled Students," *The Computing Teacher*, Vol. 8, No. 5 (1981), p. 11.

Seymour Papert, *Mindstorms: Children, Computers, and Powerful Ideas*, Basic Books, 1980.

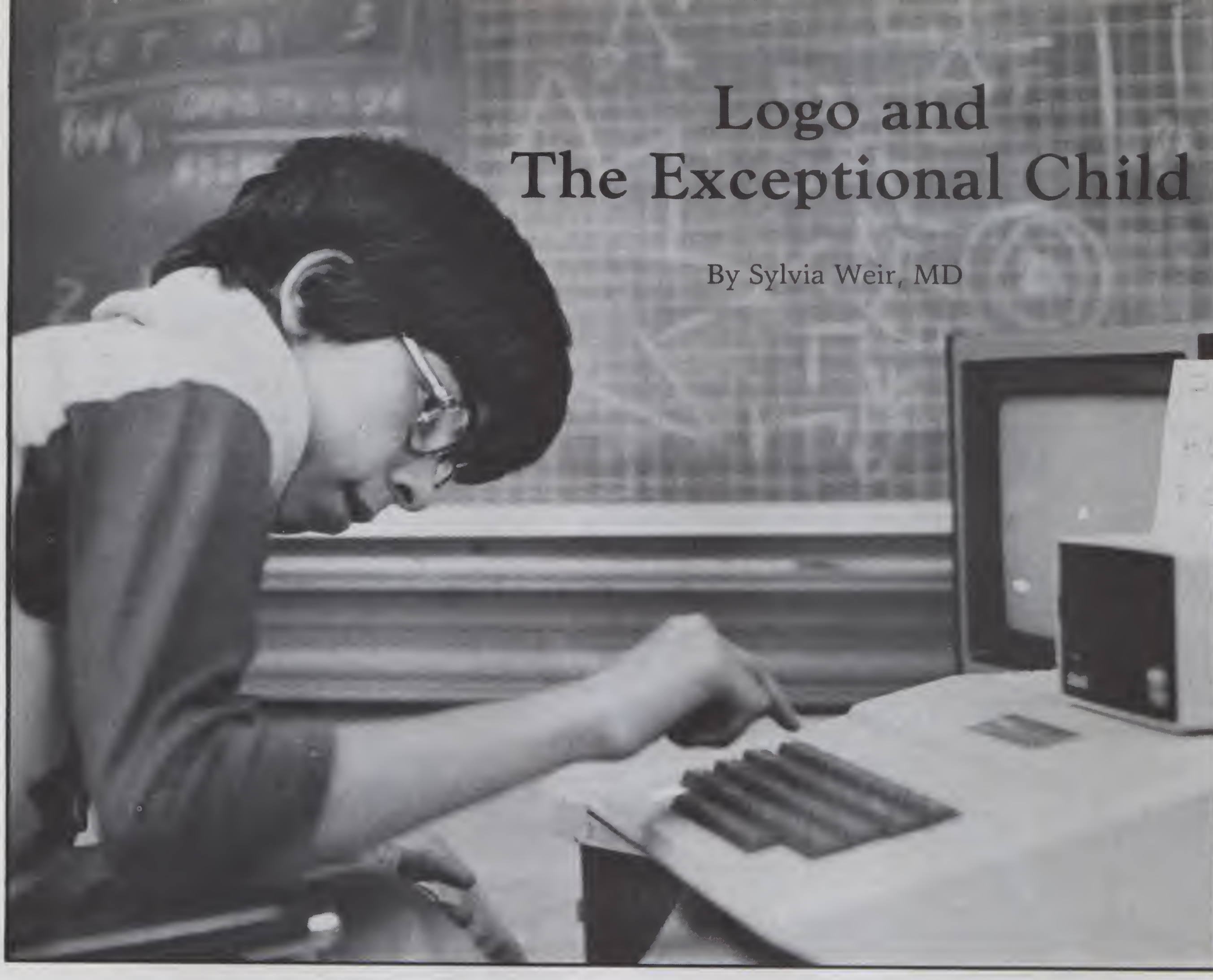
Jim Edlin, "The mass market micro: The demise of documentation," *Infoworld*, Vol. 3, No. 2, Feb. 21, 1981, p. 9.

For an excellent discussion of computer literacy see Daniel H. Watt, Computer Literacy: What Should Schools Be Doing about It? *Classroom Computer News*, Vol. 1, No. 2, Nov.–Dec., 1980, p. 1.

Marilyn Burns, "Getting Kids Ready for Computer Thinking: Thoughts for Teachers," *The Computing Teacher*, Vol. 8, No. 1, p. 28.

Theoretical Framework: The Educational Uses of Computers is adapted from Olds, Schwartz and Willie. In particular, I'm indebted to Judah Schwartz for the initial formulation of this framework. I am also indebted in this paper to my colleagues, Art Bardige and John Richards, with whom I have been preparing a more extended discussion of some of these ideas.

See List of Advertisers on page 210



Logo and The Exceptional Child

By Sylvia Weir, MD

Logo has been used with children of all ages and all ranges of ability. In this article, I focus on students who are educationally handicapped. The Logo learning environment provides a versatile tool for diagnostic, instructional and remedial use with children who have special educational needs.

It is characteristic of the Logo system to promote a close association between a child's cognitive development and an environment where he/she takes the active role in learning. This becomes especially important in work with special needs children; for example, in the work with the autistic child, with the learning-disabled child and, most poignantly, with the child who has cerebral palsy, where a particularly debilitating consequence of gross physical handicap is the dependent, passive role it imposes on its victims.

Keith has a learning disability called dyslexia. He was eleven years old when I met him. He had been a pupil at a special school for the learning disabled for four years, and had come to the clinic at Children's Hospital for assessment. He spent 40 minutes interacting with the Logo computer system that I was demonstrating to the clinic personnel. His performance was excellent—way beyond what one would expect from a child of his age, or from a beginner of any age, for that matter. In the first half-hour, he wrote a program which drew this snowman (see Fig. 1), effortlessly mastering the set of Logo primitives needed, and centering the hat on the circular head without hesitation or error. He had been labelled as having "low motivation and a poor attention span." We saw no evidence of that. All through the session his attention was totally engaged by the programming he was doing, and he had to be practically dragged away from the machine, long before he was ready to go.

The Logo system has a number of special features, two of which were especially important for Keith. Right from the start, he took the initiative—he was in control and he decided what he was going to do. The second point to notice is that he was choosing among things that he was able to do very well. That was a very unusual state of affairs for Keith, who had spent his time being unsuccessful at things other people were trying to get him to do. Not surprisingly, he lacked self-confidence and had a rather poor view of himself. His spectacular success during this Logo session arose from a happy match between his own spatial excellence and the characteristic *spatial nature of Logo activities*.

We do not yet know how many children are like Keith, children who have reading problems but are spatially gifted. We think that the group is large. These are the students who do well at games but are poor at academic subjects, as currently taught. Much of the school curriculum is language-based and does not suit the

Sylvia Weir, M.D., MIT, 545 Technology Square, Cambridge, MA 02139.

```
REPEAT [ FD 10 RT 10 ] 35 (executes almost complete circle)
PU
FD 90 (positions turtle for next circle)
TO CIRCLE
>1 REPEAT [ FD 10 RT 10 ] 36 (teaches computer to circle)
>END
CIRCLE DEFINED
CIRCLE
PD
CIRCLE (executes second circle)
PU
FD 80
BK 20
RT 90
FD 90
BK 90
PD FD 90      (proceeds to put hat onto snowman)
FD 20
BK 20
BK 10
LT 90
FD 50
OUT OF BOUNDS
FD 30
LT 90    (executed extremely rapidly; appeared to compute or guess the width of the hat to be 50 and
FD 50    then "knew" to go down 30 to complete the opposite side of hat)
LT 90
FD 30
SAVE
GO BY
YOU HAVEN'T TOLD ME HOW TO BY
GOODBYE
```

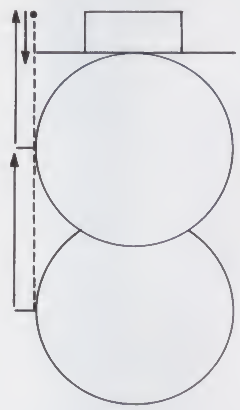


Fig. 1. The "snowman" procedure and the resulting figure drawn by the turtle.

thinking style of these individuals very well. They have difficulty with language use, and since schools do not provide an opportunity for them to display their superior spatial skill in any formal academic way, they lose out. They do well at drafting—they are good with their hands—and they are relegated to a lower track. This is where the computer graphics screen comes in. It allows us to create a world in which to do spatial problem-solving. This gives people like Keith the opportunity for academic success and for personal fulfillment.

There are many other examples of students who are educationally handicapped at present, but could benefit from a computer-based environment like Logo. This has become a real possibility for all school systems now that Logo has been implemented on inexpensive home computers. There are two such systems already implemented—a version for the TI 99/4 home computer and one for the Apple—and several others are in the making (see "Logo for Personal Computers" by Harold Nelson in the June 1981 *Byte*, p. 36).

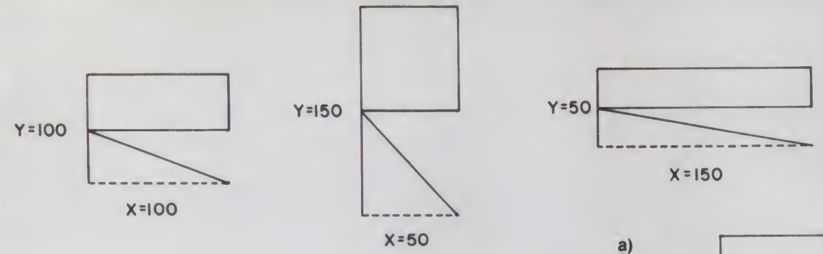
It is worth stressing one reason why this is so. Logo is good for most learners, handicapped and non-handicapped alike.

Procedural Thinking

Everybody has experienced the power of *learning by doing*—how you really get to "own" a piece of knowledge when you have used it. It is possible to make this a regular experience, even with subjects like math and science. We reverse the usual relationship between computer and student that is found in a conventional computer-assisted instruction situation. There, the clever program teaches a "dumb" student. In contrast, in the Logo system, the student is required to—gets a chance to—teach this dumb computer how to carry out the task in question.

The first step is to set up a miniature communication system linking up various devices to a computer so that the student can instruct (program) the computer to perform actions. Next we need a world interesting enough to inspire young students to want to teach the computer to carry out activities in that world. The development of the Logo language has included the invention of such domains. The best known of these is Turtle Geometry. Here the object carrying out the commands is either a mechanical "turtle," or, more usually, a cursor on a graphics screen. In either case, the turtle is capable of moving forward or backward in a particular direction relative to itself and of rotating about its central axis. As you command the turtle to move around the screen, your intuitive knowledge about how you move your own body around in space is mobilized. This gives you a way of understanding a whole range of mathematical ideas by giving you a "feel" for what is going on.

Photo 1. Dr. Weir discusses a Logo project with a Cotting School student.



These attempts to have the turtle draw the letter R led the student to investigate some basic notions of practical trigonometry. These concepts were presented in the following form:

a. identify the angle whose value you want to find

b. define the rectangular triangle that has as one of its angles (different from 90°) the angle you identified in a.

c. in this triangle identify the sides that are adjacent and opposite to the angle and find their values.

d. the value of such an angle is given by:

$$\text{ANGLE} = \text{ARCTANGENT}\frac{\text{value of opposite side}}{\text{value of adjacent side}}$$

e. the length of the biggest side of the triangle (hypotenuse) is given by:

$$Z = \frac{\text{value of adjacent side}}{\cos(\text{ANGLE})}$$

This procedure could then be used to draw other letters with sloping lines.

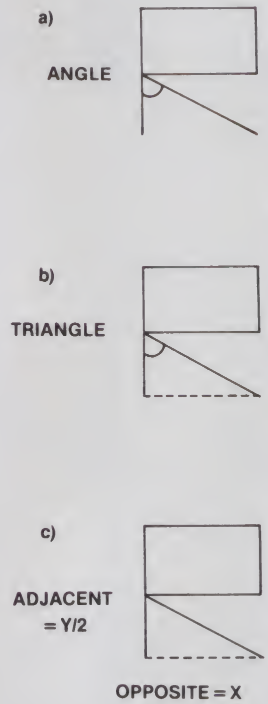


Fig. 2.

This brings us to a central idea. The *meaning* of a square is captured by the procedure which draws the square. When one thinks of the square in this way, that is to say, when you are engaged in procedural thinking, you can have an understanding of the meaning of the square in terms of how you produce it and how you might want to use it. For instance, think of the angle between two lines as the amount you would need to turn around if you were to do the following:

1. Start off facing the direction of one of the lines.

2. Turn on your heel until you are facing in the direction of the other line.

Understanding an angle as a change in direction gives meaning to angles; for example, the special angle of 180 degrees is the amount you turn to face in the direction opposite the one you are facing now; 360 degrees of turn gets you all the way around.

Thus when a Logo student is learning to program, that is not the only thing he is learning. Most particularly, the activities are designed to serve as a tool for the understanding of considerable portions of classical mathematics and physics, from the idea of angle and regular polygon, up through concepts underlying differential geometry and many classes of differential equations, including the equations of motion of Newtonian physics.

Most drawing projects lead quite naturally to interesting math. For example, writing a program to draw a letter of the alphabet, allowing variable size, generated a bridge to practical trigonometry (see Fig. 2).

Next I describe work with Logo and physically handicapped individuals.

Trapped Intelligence

Cerebral palsy is a disorder of movement and posture resulting from a lesion in the brain occurring before the end of the first year of life.

Mike was 17 years old when he first met a computer. He has cerebral palsy involving all four limbs, more marked on the left side. His speech is severely affected, but he can be understood with difficulty. He has sufficient motor power to control his wheelchair, but has never used a pencil. Mike has spent six to nine hours a week at the computer since the onset of the project, has been brought to the Logo laboratory weekly during vacations when the computer at the school was not available to him, and lies in bed solving Logo problems in his head. He has become a competent computer programmer, and has just been accepted into the Computer Science course at the University of Massachusetts, Boston campus.

Mike is one of several students who have participated in a research project over the past two years at the Cotting School for Handicapped Children. We

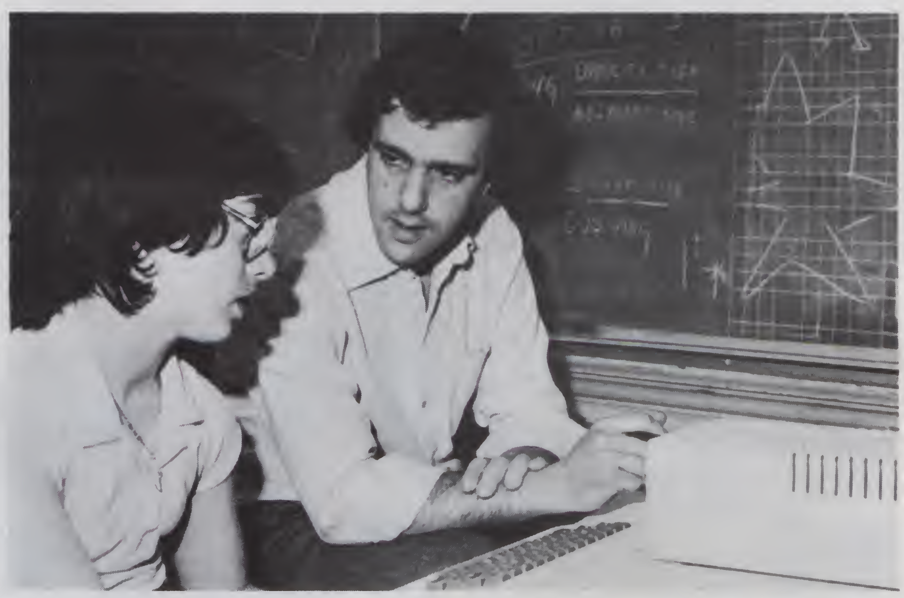


Photo 2. Jose Valente, research assistant on the project, assists a student with one of her procedures.

✓See List of Advertisers on page 210

were interested to see how Logo would fit into the curriculum of physically handicapped students. The results have been dramatic.

Introducing computers into the learning environment of severely handicapped students can revolutionize their lives. Students whose intelligence has up till now been trapped because they cannot communicate what they know—either because they cannot write, or because they cannot speak, or because they can do neither—and who in the past would have remained totally dependent, financially and otherwise, on others, can now look forward to a measure of independence, to the possibility of earning a living, to the possibility of enjoying a future with dignity.

Imagine you have grown up having never used a pen or pencil. You are capable of sophisticated intellectual activity, but because of your handicap you have to do it all in your head. *Someone comes along and offers you a chance to "write it down."*

You are accustomed to sitting around waiting for someone to be free so that you can get on with your learning, which is only possible with them around.

Someone comes along and arranges things so that you can now get on with it, when you want to, without disturbing anyone else, at your own pace, and for as long as you feel moved to do so.

You are used to someone else taking the initiative, while you sit around passively. You don't really expect any action that you take will make any difference to the world out there. Things get done to you, rather than get done by you.

Someone comes along with the most powerful piece of technology known, and puts you in control of that magic. It is you who gets to choose what project you will do, what you will "teach" that "dumb" computer to do.

Now you can:

- Use the computer as a notebook and as a scratchpad—no longer is it necessary to do all your schoolwork in your head.
- Use the computer to do sophisticated mathematics and physics.
- Learn computer programming, including business programming.
- Do all your language arts assignments on the computer.
- Engage in problem-solving tasks, especially involving spatial reasoning.
- Do music theory and composition.

A Tool for Diagnosis and Remediation

Several features of Logo make it suitable for both diagnostic and remedial work with physically handicapped children who have learning problems.

An important element in the success at Cotting School is the way problem-solving experience is built in to the daily Logo activity in a very direct way, such that the student can become an "embryo scientist," making hypotheses about likely outcomes of one or another course of action, testing these ideas, and refining the hypotheses in the light of experience. For these handicapped students, working on a computer gives them

their first experience of tackling problems where *they themselves* initiate solutions, try them out, decide when to change track and when to persist, respond to feedback, and so on. All those things tend not to happen in the dependent adult-dominated life of a handicapped person.

An interactive computer graphics system lends itself to an emphasis on process rather than on product. The intermediate steps in the process of solving a problem are clearly displayed—out there on the graphics screen for both the learner and the teacher to see—more clearly than if he were performing an internal task. This shows up any discrepancy between what is intended and what actually happens. Any error or "bug" can become a source of insight into the particular learning process of the child.

All the children involved have been able to type, albeit slowly, by dint of a variety of ingenious props. The delete key is invaluable since they do hit unintended keys. Exercising the affected hands and fingers is a side effect of all the typing, and has resulted in unexpected improvements in fine motor control. We are designing a multiple input communicator for children who are too disabled to type.

If these severely handicapped individuals are to hold down a job, the quality of their writing is important. The word-processing capability of the system gives these students their first opportunity to generate their own text, without the intervention of an adult. It is also the first time that their teachers have been in a position to assess their written language ability. Some students now do all their language arts work on the computer. A typical improvement is shown in Fig. 3.

Just as we did with Keith, so we can exploit the spatial nature of Logo activity in our work with the physically handicapped. This time our aim is to remedy the weakness in spatial understanding which results from the

"I ment Dr. Sileva Where, Jose Valente and Gary Drescher on October 5, 1978 at 9 : 32 : 47 AM. which the compuer I was so excized it like being it a waitting & maternace room at a hospiltal whiting to fine it oot's a boy or a grail.
We had a and we whont you to do it fist for us I am LOGO number "1" ginny pig.
When they get a new idse they say to hel, michalel we had a and we whont you to do it fist for us

* * *

My name is Michael Murphy. I am the person whom your mother saw on "PM Magazine." I attend the Cotting School in Boston mass. I have been work with the compuers for about two and a half years. The name of the system is "LOGO". It has open many new doors for me. Now I can draw pictures on a cheen ane write letter like this one.

Fig. 3. The first example is taken from a letter Mike wrote when he first entered the Logo program. The second example is from a letter written earlier this year.

See List of Advertisers on page 210

The teachers have noticed
the turnaround in their students,
the increase in confidence,
the improvement in self image.

unusual childhood of an individual handicapped from birth. Development of an understanding of spatial concepts depends on information generated by handling objects and moving around in space, with the coordination of several kinds of information—tactile, visual, motor and kinesthetic (sensation in muscles and joints of the moving part)—giving a description of the direction and amount of movement.

If the handicap is sufficiently severe, the child will not have handled objects as part of growing up and will not have played with blocks as an infant. Our Logo environment can begin to fill the gap. Lines on the graphics screen can become models of objects, and manipulations of such graphics objects can supply manipulatory experience of a sort, involving a minimum of motor effort—just the pressing of a key.

Furthermore, it is difficult to test the performance of severely handicapped persons, given their motor disability. Again the computer can help. We have extended the standard Logo system and tested our handicapped children on graphics screen versions of various standard assessment tasks. [**Note:** A substantial programming effort has produced an efficient general-purpose system for implementing a wide variety of tasks. New tasks can be added with relative ease. Features of the system include an automated record of the moves made by the subject, complete with a time marker, and a facility for using this record to drive a rerun of the task for purposes of analysis.]

We are now able to combine the results from these screen assessment tasks with information derived from standard Logo projects to make fine-grain analyses of the ability of our students. Together with the teachers, we are evolving remedial activities tailored to the needs of each particular child. Support materials include "games" designed to build on strengths and exercise weaknesses.

Delay or a deviation in the development of cognitive ability can be associated either with brain damage or with lack of a particular type of life experience, or with both. Irreversible brain damage places a constraint on the level of improvement possible, but we have been surprised by the amount of learning that can take place when appropriate experience is supplied.

Motivation

A high degree of motivation offers the best chance for significant learning. An increase in motivation among our students is reflected in an intense competition for the use of machines. Students come in an hour before school begins or skip half the lunch break to work on the computer. Animated discussions about various computer projects are frequent.

This should not surprise us. When you have to find out something to achieve some goal you have set yourself, your motivation is high and you're likely to go after it. Since the literal computer cannot understand nearly-correct instructions, as humans do, students develop a respect for accurate spelling during the process of inputting their programs.

The sheer fact of this whole new world of knowledge and of their own independent management of this has had a profound effect on the intellectual life of these students. The impact on the institution has been far more than just the introduction of another subject into the curriculum. The teachers have noticed the turnaround in their students, the increase in confidence, the improvement in self image. They have noticed this sudden access to all kinds of problem-solving activity that was not possible before.

The school authorities have been so impressed by all this that they have raised $27,000 to equip a computer center. Computer programming has become an integral part of the school curriculum and 41 out of 50 senior students have opted to take computing. Four of the seniors involved in our project have changed their career


OKIDATA
365
Announcing the 80-column Model 82A quality Microline printer. Quiet. Long lasting. Exceptional print quality.
• 120 cps
• 9-pin head, one year warranty
• Bidirectional, logic seeking
• Graphics
• Parallel and serial interfaces—standard
• Friction and pin head feed—standard
• And much more
From Okidata. The company that stands out. Apart from the rest. For more information on Okidata's incomparable line of printers contact us at 111 Gaither Drive, Mt. Laurel, NJ 08054, (609) 235-2600.
μ 82A microline
FORM LENGTH
SEL PAPER POWER
TOF SET
SEL
FORM FEED
LINE FEED
Making small printers for people who think big.
Okidata is subsidiary of Oki Electric Industry Company Ltd.

plans to computer programming, and have been accepted into college computer science courses. Increased vocational access—the possibility of independent living—is the great advantage accruing from the computer activities.

Logo Opens Another Kind of Communication

The last category of special needs I will talk about is autism. The fascination for machines shown by autistic children can be exploited in a Logo-based learning environment. Active and enjoyable explorations in controlling the Logo turtle can form the basis for the development of language for communication, both verbal and nonverbal. We used a mechanical turtle which moved around the floor and a simplified input device called a button-box, where pushing a single button had the effect of typing a primitive Logo command, for example, FORWARD 30.

Donald was a seven-year-old autistic child who worked on the computer for seven sessions over a period of six weeks. Each visit lasted about an hour. The one-to-one correspondence between pushing a button and getting a single response from the turtle turned out to be the magic ingredient! Donald began to predict events on the basis of a growing understanding of the relationship between his button-pushing and the turtle's behavior. He created acting-out sequences, demonstrating the connection in rather an explicit way between his own movements and the turtle actions. He made action speeches—pushing his own belly button, saying "up," standing up, pushing PENUP on the button-box, pushing his belly-button again, saying "down," sitting down, pushing PENDOWN on the button-box. He began to vocalize his thoughts spontaneously during this "turtle play." He volunteered "me drawing"; "turtle goes backwards and forwards"; and later, "see how it works!" Unlike the private nature of his usual monologue, these spontaneous turtle utterances made sense to us.

Neither the use of spontaneous language nor the active seeking out of social interaction which Donald displayed during his Logo sessions had been observed prior to this experience. We speculate on the reasons for this felicitous outcome as follows:

Part of learning involves knowing what to record in the situation. In a complex setting it is difficult to see what is relevant and what is not. One of the crucial features of Logo is the way it is relatively easy to isolate the salient features because these correspond to the only elements in the turtle's state which the child can change—the position of the turtle, its heading and the state of the pen. *Thus the child knows where to look.*

This is good for all children, and most especially for autistics, who have a notoriously idiosyncratic sense of relevance. The advantage of using Logo is to provide a shared relevance explicitly. Sharing a sense of relevance leads to a shared understanding that can form the basis of communication. This possibility is enhanced by the specific activity in a Logo situation. Controlling the turtle embodies the idea of communicating a sequence of commands in order to get the turtle to achieve a goal. Controlling the turtle can act as a model in terms of which to grasp the idea of communication.

Teachers and the Microcomputer

I have tried to convey the enormous potential of the microcomputer as a versatile and flexible educational tool for teachers. This is true for all teachers, but most especially for those working with exceptional children. I am impressed with the potential of the microcomputer as an aid in the provision of least restrictive environments for children with special needs.

Experienced teachers can react in one of two possible ways to the arrival of the computer on the education scene. "Do I have to get involved with that stuff?" or "I was on the verge of giving up! Burnt out! And then I got involved in this whole new fascinating world."

For some teachers, it is the long awaited opportunity. For others, it feels like the beginning of the end. If too many of those in leading positions in education fall into the latter category, the effect could be nothing short of disaster. The reason for concern is clear. The last thing the experts in the educational field should want is to leave decisions about this whole new area to "outsiders"—to non-educators. The risk is that we will continue to get the kind of computer aids to instruction that computer experts want to produce, rather than the things educators want. This will only perpetuate the view, not entirely unjustified, that much of what happens when computers are involved in education is pedagogically retrogressive.

I would hope to see teachers becoming more and more involved in deciding the directions that the microcomputer revolution should take. I offer Logo as the most congenial means currently available for making that possible. ■

Controlling the turtle can act as a model . . .
to grasp the idea of communication.


OKIDATA
53
Announcing the 136-column Model 83A quality Microline printer. Quiet. Long lasting. Exceptional print quality.
• 120 cps
• 9-pin head, one year warranty
• Bidirectional, logic seeking
• Graphics
• Parallel and serial interfaces—standard
• Friction and tractor feed—standard
• And much more
From Okidata. The company that stands out. Apart from the rest. For more information on Okidata's incomparable line of printers contact us at 111 Gaither Drive, Mt. Laurel, NJ 08054, (609) 235-2600
µ 83A
microline
Making small printers for people who think big.
Okidata is subsidiary of Oki Electric Industry Company Ltd.

Examining the gains—both intellectual and monetary—to be made from the use of micros in the classroom.

Educational Computing—The Giant Awakes

By Lloyd R. Prentice

Let's look at the stakes. According to Market Data Retrieval, there are 83,334 public schools, 21,749 nonpublic schools and 3453 colleges in the United States. There are 1,770,217 public school teachers and 477,281 college faculty. There are 121,968 school district personnel, 39,987 public school librarians and 27,510 college administrators. There are nearly 47 million elementary and secondary pupils. The elementary and secondary schools alone spend nearly $2.5 billion a year on instructional materials.

These figures don't include the market for educational materials in the home; the export market; business, professional and industrial training; or continuing education.

You don't need VisiCalc to see what these figures mean to computer hardware and software vendors. Imagine banking the revenue from the sale of one $500 administrative package to every principal in the country. Or achieving a 40 percent penetration of all classrooms with a $1500 microcomputer system. Or the rewards from capturing the market with a major reading software package. Or the royalties on a best-selling book that teaches computer literacy to teachers. Such fantasies make the mouth water.

But the education market is notoriously difficult. Educators are penny-pinchers, and for good reason—their budgets are tighter than a fiscal conservative in a voting booth. Educators strive to cut margins to the marrow. They demand prodigious service from the vendor. They subject products to the most gruelling environmental testing imaginable—the classroom. They are not known for innovation and their purchasing practices are often highly politicized.

The market in education for hardware and software is particularly difficult. Before computer-related products can be sold in appreciable volume, the educators must be educated—the administrators must be instructed in the cost/benefits of educational computing and the teachers must be taught how to teach with computers. And, most important of all, considering that education budgets are already at survival level, the dollars must be squeezed out of somewhere at the expense of something else.

So, while the potential market for computer-related products in education is mouthwatering, the question is, will this market truly fulfill its potential and, if so, when and how? And more, who will ultimately profit?

The Promise of Computers

The pedagogical promise of computers has been known since the 1960s. In those years of The Great Society, lavish government funds encouraged important formative research at such institutions as Stanford University and the University of Illinois. Out of this research emerged a number of commercial computer-assisted curriculum packages designed to run on timesharing mainframe or mini systems. These packages, offered by such companies as Control Data Corporation and Pat Suppes' Computer Curriculum Corporation, proved quite viable from an instructional point of view, but as government funding tapered off, only the most affluent and innovative districts were able to justify them.

As recently as a year ago, most educators were unaware of the classroom potential of microcomputer technology. Of the few who were aware, many felt threatened by the technology. Said one Massachusetts teacher shortly after the passage of that state's tax-slashing Proposition 2 1/2, "How can I get excited about computers when one out of four colleagues in my school will not be rehired next year?"

Despite an initially low level of awareness, tight budgets and a certain level of apprehension, the 1980/81 school year was a remarkable year for educational computing. It was the year that the personal computing revolution of 1975/76 finally caught up with the conservative institutions of education.

Computers were a hot topic in edu-

Lloyd R. Prentice is publisher of the Classroom Computer News, *PO Box 266, Cambridge, MA 02138.*

cational circles. The year unfolded like one massive crash course. The National Council of Teachers of Mathematics came out with a very positive position statement and several important publications. The American Federation of Teachers set up a large educational computing display at their annual meeting. The National Education Association set up a teacher-training program. The American Society of Curriculum Developers offered a special presession on computer-assisted instruction at their annual meeting. The National Council of Teachers of English appointed a special technology committee.

But these were just the icing on the brownie. The traditional education journals published scores of articles on computers. Several magazines devoted exclusively to educational computing were born in the 1980/81 school year—including *Classroom Computer News*. Hundreds of in-service training workshops were held across the country and thousands of computer committees were appointed in schools and in parent/teacher organizations. At least four college degree programs in educational computing were launched in 1980/81—including one at prestigious Stanford University.

Now, as we move into the 1981/82 school year, talk to an educator about computers and chances are better than even that she'll tell you all about Apple and Commodore and Radio Shack and bytes and RAM and ROM.

Says Dr. Ludwig Braun of the State University of New York at Stonybrook, "I have been in this field for 15 years. In the last 12 months there has been an enormous increase in the level of activity. Now, just this year, at least three school districts on Long Island have more than 100 computers in classrooms. Moreover, I used to get one or two requests for presentations a month—now I get one or two a week. And not just from teachers—now I'm getting requests from district *superintendents* and department chairmen as well."

None of this has escaped the notice of the hardware and software vendors. Indeed, much of this activity was catalyzed by forward-thinking firms. Apple set up a private foundation and gave away $150,000 in equipment to educators with creative ideas. Bell & Howell won a massive 600-computer bid from Texas, largely on the strength of an aggressive in-service teacher-training package. Commodore gave away free computers—one for every two purchased by a school. Radio Shack offered free computer literacy courses to teachers, who were taught in local Radio Shack stores. Most of the traditional textbook publishers announced computer-related products, although one industry observer says that, with two or three notable exceptions, there was more fanfare than delivery. Meanwhile, a dynamic industry of small educational software producers flourished with start-ups, success stories and quiet fadings into the night beyond count.

As of October 1980, 52,000 microcomputers and terminals were available to pupils for instructional purposes in American schools.

How Many?

It's not easy to get reliable figures on the size of the educational computing market—not least because it is a fast-moving target. Various figures are bandied about, but quote any published figure and listen to somebody scoff.

The National Center for Educational Statistics says that as of October 1980, 52,000 microcomputers and terminals were available to pupils for instructional purposes in American schools, with microcomputers outnumbering terminals three to two. The NCES survey also indicated that over 30 percent of the schools that did not offer instructional access to computers at the time of the survey planned to do so within three years.

Glen Polin, marketing manager for education at Apple Computer Corporation, says that the NCES survey substantially undercounted the number of computers in the schools. Many computers come in through the back door, he says, purchased by parent-teacher organizations or under local budgets as "audio-visual devices." As a result, the higher-level administrators who fill out national survey questionnaires often don't have an accurate notion of how many computers are actually in use in their schools.

"I talked with the data processing director at one college," says Polin, "who swore that there was only one Apple in the whole institution. He knew because he'd bought it. But I knew from warranty cards that there were at least seven Apples at work on campus."

Polin says that sales to educational institutions represent 25 percent of Apple's business and that that percentage is increasing. Apple claims more than 200,000 units sold to date, leading one to speculate that some 50,000 Apple computers have been sold to schools. Radio Shack has sold that many or more to the education market, according to industry sources, although Radio Shack says that they have no way of really knowing. According to Chuck Carlson, a publisher at Random House, Radio Shack and Apple share about 80 percent of the educational market for small computers, with Commodore, Atari, Ohio Scientific and others competing for the remaining share.

Creative Strategies, a California market research firm, projects that the combined retail value of microcomputer shipments to educational institutions will exceed $1 billion from 1980 to 1985. They also project that the share of the total market for microcomputers going to education will more than double over the period. Glen Polin lends credence to this projection by saying that Apple sales to education have increased more than 100 percent a year over the past several years. Bell & Howell, which markets a specially modified version of the Apple II to educators, is reluctant to discuss figures, but according to educational marketing manager Frederic Michael, Bell & Howell has experienced a growth pattern that is similar to that reported by Apple, and that his company's projections are on the high side of those reported by Creative Strategies.

If it is difficult to get a fix on hardware sales, getting a fix on software sales to education is even harder. One executive projected a $75 million to $100 million total market for educational software this year. Another

doubted that the market would exceed $2 million. And still another executive said, "Regardless of what you hear, everybody is lying."

But regardless of the numbers, there is near universal agreement that software is a major sticking point in the educational computing field. Vendors question what will sell in the schools, how to produce it most effectively and how to protect it from unauthorized duplication, while educators question the quantity and quality of software available today.

There is near universal agreement that software is a major sticking point in the educational computing field.

To What Ends?

Just how can computers serve the process of education? If we think of the computer as a general-purpose tool for working with symbols such as numbers, words, images and sounds, we realize that the possibilities are endless. At present, several major applications stand out:

- As an instructional management tool—keeping track of grades, homework assignments, progress through a course of curriculum, special education plans;
- To teach computer literacy—preparing students for a computerized world;
- For drill and practice—permitting students to review facts and basic skills at their own pace and in their own time;
- As a tutorial medium—the computer, as a lineal descendant of the teaching machine, can present information in textual and graphic form, quiz the student and branch depending upon student response;
- As a discovery tool—programming languages, word processors, simulations, games, graphics packages and mathematical modeling routines all provide tools that help students explore and create ideas. The difference between these applications and the tutorial applications is that here the student takes the initiative in the learning process, rather than the computer.

Solid research on the effectiveness of the computer in the classroom is still sketchy. But the current consensus is that the computer can accelerate learning in many situations. There is no evidence that a computer can substitute for a good teacher, but there is evidence that a well-developed computer-assisted curriculum can help less-able or less-experienced teachers do a better job.

George Hanify, director of computer applications at the Merrimack Education Center in Massachusetts, for example, has been evaluating the effectiveness of timeshared reading curriculum in several Massachusetts schools. One year he found few gains in the classes with computer instruction relative to control classes. The next year he found significant gains. The difference seemed to be that in the first year the computer system was up against a talented, experienced team of teachers. In the second year, the teaching staff instructing the control group was much less experienced.

"The computers seem to have helped relatively inexperienced teachers get up to speed much faster," says Hanify.

Many educators feel that the potential of the computer as a learning tool has yet to be tapped—we simply don't know yet how best to use it.

Says Dr. Ludwig Braun, "If we give youngsters spontaneous access to computers, the results will be staggering. Children who are now three to five years old, if given the kind of access to computers that is now possible, will develop intellectually in ways that we cannot now comprehend—with depth and breadth beyond anything we can conceive."

But instructional effectiveness is only one dimension. Two other benefits are of nearly equal importance. First, computers can help take a heavy management load off burdened teachers and administrators; and second, kids love them.

Dr. Kirk Wilson of Boston University says that in the area of special education alone the management load in education is staggering. The 1975 Education for All Handicapped Children Act, for example, requires that handicapped children be serviced individually and "integrated" into regular education wherever possible.

"Special education teachers must now prepare annual individual educational plans of two to 25 pages for each of this country's 4.03 million handicapped students," says Wilson. "The (plans) must state present levels of performance, specific individualized annual goals and short-term objectives for the coming year."

Wilson, through a company called Learning Tools, Inc., has developed a sophisticated microcomputer-based management system to help educators out of the resulting logjam.

The management potential of the computer helps sell the teachers and the administrators, but the fact that the computer is fun to use, patient, never moody, picky or cross and that it encourages interaction sells the kids. Chuck Carlson of Random House tells of a school that installed computers with rather conventional tutorial software—the purportedly dull basic reading and 'rithmetic stuff. The students liked it so much they pressured the administration to open up the school an hour before and an hour after regular hours so that they could get extra time at the computers.

One teacher says that at her school the kids are in the habit of stealing anything that isn't nailed down. "But they don't rip off the computers," she says. "They know that if they do, school will be a much more boring place."

Selling Points

While hardware and software vendors note the exploding enthusiasm for computers in the schools, however, they also wonder when the interest will translate into sales and profits. The hardware manufacturers seem less worried on this score than the software people. According to Bell & Howell's Frederic Michael, the educational computing market is still hardware-driven and, despite tight money, resources are being diverted from many sources to bring computers into the schools.

The major concern of hardware vendors is the likely effects of Reagan administration cutbacks on federal funds to education. The present National Science Foundation (NSF) budget for science education is $130 million, according to NSF's Dorothy Derringer, but the Administration is proposing $9.9 million for next year and the most promising proposal from Congress is $75 million. A large share of this budget goes into computer hardware and software.

✓See List of Advertisers on page 210

U.S. House of Representatives science consultant Robert Smythe sees "serious dislocations" in educational computing programs in the event of a substantial withdrawal of federal funds. Several observers feel, however, that the dislocations will be more serious in the long term than the short. Schools have found many nonfederal sources of funds for computers—one school sells software; the students of another organized a fund-raising program. Bake sales are a popular way to raise funds for school computers, and in Bartlesville, OK, the PTA sold candles.

The long-term effect of federal cutbacks, however, will show up in a growing disparity of educational advantages between affluent and poor school districts. The affluent districts will find the money to continue vigorous computer-aided programs, but the poor districts will not. Moreover, a cutback in research funds will curtail some important research and development projects, according to Dorothy Derringer.

"We'll be stuck in the drill-and-practice mode, but the more imaginative applications will develop much more slowly," she says.

> If given the kind of access to computers that is now possible, (children) will develop intellectually in ways that we cannot now comprehend.

The software vendors are a slightly more pessimistic lot than the hardware people. Chuck Carlson of Random House points out that there is still not a large enough base of hardware in the schools for major traditional publishing houses to justify enormous development funds for software. Several large publishers, including Science Research Associates, Scott-Foresman, Random House and Addison-Wesley have made moves in that direction and several others have made noises. But medium-sized Milliken seems to hold a position of leadership for the moment in product availability and commitment to development.

Some people say that the large publishers are too conservative to carry the banner, however. They expect new faces to show up among the leaders of educational publishing as smaller and more aggressive companies get the jump on the traditional giants in product development. But even smaller companies have questions about the potential of the educational market.

Says John Victor of Program Design, Incorporated, a small but aggressive software developer, "The education market has not been very regular. The schools have been stealing software and that's not a very good situation for developing software."

The issue of unauthorized copying or piracy in education is a tender topic. Most publishers have horror stories, but few publishers are willing to take on a large school district in a

court of law, particularly one that may very well be a major customer for other materials. Consequently, there is evidence that some publishers have held back on investment in software development while awaiting a solution to the problem.

However, Rick Holden of J. L. Hammett, a major school supply house, says that "too much time is being spent on trying to protect the product and not enough time is spent on creating good curriculum materials." Chuck Carlson of Random House concurs: "My boss says that we as publishers cannot be concerned about a small percentage of people who are going to do something wrong."

Allen Rosen of Microphys Programs, another small educational software developer, says that while piracy is a problem, the fact that federal funding has to date subsidized public-domain software development jeopardizes return on investment from the private sector, and that this has held up development. It is true that thousands of programs are being written by teachers and students in schools and many of them are excellent. But the difference between a program and a marketable product is profound.

Quality marketable curriculum materials must be educationally sound, appropriate for a large number of users and technically polished. This argues against development by the inspired amateur and for development by well-integrated teams of curriculum specialists, programmers, writers and artists.

But the reason that educators don't yet have the quantity or the quality of software that they want is probably due neither to piracy nor competition between federal and private resources. Chuck Carlson points out that the market for microcomputer-based materials in education is really less than a year old.

"I doubt if this year the market will live up to its expectations," he says. "A lot of hardware will have to be bought before we'll have a larger software market. It takes almost three years to do anything and we're just getting started."

Other people concur. Cheryl Wiener of McGraw-Hill notes extraordinary increase in the level of activity within her company over the past year. She points to quite a number of companies that are gearing up for release of products in the 1981/82 school year.

The educational market seems to be coming out of a long sleep. There is formidable momentum and a notable degree of acceptance of the computer by administrators, teachers and students. Moreover, there is substantial outside pressure being exerted on schools to expose their students to computers, in order to prepare them for a world that is increasingly run by computer.

The two major forces that may slow up and stand in the way of the development of the educational market are federal cutbacks and the massive job of training more than two million administrators and teachers. But if last year is any indication, our educators are up to the challenges. If the educational market does indeed develop to its potential, the indisputable winner will be society as a whole.

Dr. Braun of the State University of New York at Stonybrook says, "Put a computer into the hands of a kid and then get out of the way." It looks like that is just exactly what we as a society are setting about to do. ■

✓See List of Advertisers on page 210

You won't lose your cool with this lesson-planning program in Microsoft BASIC.

Tame That Blackboard Jungle

By James A. Ingram

There I was, faced with the beginning of a new school year and the many preparations that go with teaching. Dismally, I recalled former years of lesson plans scribbled on the prescribed forms, out of step with actual class activities by midweek, hardly ever up to date, and seldom kept on file for future reference.

Trying to turn the tide of frustration, I brightly reminded myself that it was only July, and I was already finding time to think about such things. Would this be the year that I would finally get my act together?

The thought of constantly rearranging plans and the paperwork involved rang a bell somewhere in the cobwebby depths of my summer-vacation mind. Didn't I have a better way of handling such data?

Data! The computer!

I decided that a few minutes of programming before actually sitting down to write my ninth grade earth science plans would pay off in organization if not in time. Off I went to the computer.

As is so often the case with amateur computer programmers, the few minutes of programming turned into a two-day marathon. But the end product was worth it.

Designed to run on my Heath H-8 system with H-9 terminal (modifications for the H-19 are straightforward) and H-17 dual disk drives, the program performs three main functions: entry of activities, insertion and deletion of activities and display of current information. (*Even if you have no use for lesson-planning, you may find the screen display concepts useful in other applications.*)

Address correspondence to James A. Ingram, 804 N. C St., Broken Bow, NE 68822.

The program operates on a disk file that contains a list of the day and date of every actual class throughout the school year. In my school calendar, the first day is Aug. 25 and the last day is May 29. Because of Microsoft BASIC's internal operations, the file is made of two entries—the first is the day and date, and the second is a string of 39 blanks which will eventually contain that day's activity. The program dimensions for 300 days and dates and for 300 activities. Most school years are less than 185 class days, so there's plenty of room in the program for expansion. This requires quite a bit of both string and disk space, so the program uses the CLEAR 12000 statement to open the necessary string space.

The initial disk file contains blanks where assignments will be stored. It only grows if you add more assignments past the end of the calendar (more about this later). On my system each file occupies about 50 disk sectors, or about 12K—185 lines of 80 characters each. I needn't worry about it since my system has 56K, but it could be pared down for smaller systems by reducing the amount of space dimensioned or even by breaking the year into semesters or quarters. The program itself occupies only about 6K of memory in Microsoft's condensed format.

The blank calendar is the only required disk file in addition to the program and is named SY1:CLASDAYS.DAT (the disk drives are referred to as SY0: and SY1:; the .DAT identifies the file as a data file). I created it using my text editor, which generates files in the same format as BASIC data files (standard ASCII format). If you don't have this option, you can write a program using READ and DATA statements to generate the file. Other files—actual lesson plans—will be added to the disk catalog as you generate them with the main program presented here.

The program itself contains nine distinct processes. The two most important are assignment entry and screen display; in addition, it provides routines for printing, insertion of assignments between existing assignments, deletion of assignments with closing of the resulting gaps, storage and retrieval routines for the disk system, generation of completely new files and a brief listing of available commands in the event you forget them.

Program Commands

The simplest commands are single letters followed by a carriage return. Typing S will store your lesson plan on disk; typing R will retrieve it from the disk for more work. Entering N

will pull the blank file SY1:CLASDAYS.DAT into the program so you can start a new file from scratch, and typing H or HELP will list several pages of abbreviated instructions. Typing L will, among other things, bring up a screen display of the lesson plan you are working on.

The remaining commands need numbers with a letter to specify locations within the lesson plan. Typing E(15)EXAM—CHAPTER 1 will put EXAM—CHAPTER 1 into the plan at day 15. (The days are numbered consecutively for easy reference; they do not correspond to the dates.) If there was already an assignment there, the old one would be erased and lost. This is an easy way to swap one assignment for another if you change your mind for a given day.

Entering O(41,6) moves all assignments after day 41 down six days on the calendar, creating an opening for more assignments without deleting any of the original ones. It always must have two numbers in the parentheses separated by a comma.

Typing C(42) will close the gap created with the O command above. The lesson-planning program determines how many blank lines are in the gap, and automatically moves the following assignments up.

Note that in this case the number is that of the first blank, not the last assignment before the gap. If you have multiple gaps in the plan, you will have to use this command once for each gap. Only one number is used in the parentheses here.

P(16,20) gives a printed copy of the lesson plans for days 16 through 20, inclusive. Since my printer is tied to a terminal rather than a separate port, the routine gives me a chance to turn on and adjust the printer first. Then it prints the plans for the requested days. This command requires two numbers separated by a comma.

The Video Display

The screen display routine, or lister, is one of the most interesting features of the program. My system transfers data to the screen at 9600 bits per second (bps), so filling the screen with 12 80-character lines takes only about one second. If your system is slower, this may become a bottleneck in the use of the PLANS.BAS program; it should be okay down to about 4800 bps.

The display routine creates a *window* for the data one *page* in size; in my case, a page is ten lines (ten class days). The window can be moved up or down one page at a time or one line at a time anywhere within the 300 dimensioned days. This is done by typing single-letter commands. No carriage return is needed; the screen response is immediate. Line 1480 of the program (Listing 1) accomplishes this with an INPUT$(1) statement (the 1 refers to the number of characters to accept before execution of the command). Other versions of BASIC may use INKEY$ or another similar statement. The window commands are:

- F—moves window ahead (to a later date) one page
- R—moves window back (to a previous date) one page
- U—moves window ahead (to a later date) one day
- D—moves window back (to a previous date) one day
- T—moves window to first page (top)
- B—moves window to last page (bottom)
- E—exits to main command loop without erasing screen

To alter a lesson plan locate the information to be altered by using the lister, then type E. When the prompt appears, enter your alterations with the main commands. (Note that within the lister, the prompt does not appear. Commands which make changes in the data require the prompt to be present.)

You enter the lister by typing L. The result is a display of the first page (the first ten days). If you wish to enter at some other page, simply type the number of the first day desired in parentheses after the L. L(120) will

```
DAY 1   25 AUGUST 1980 MONDAY      GETTING STARTED
DAY 2   26 AUGUST 1980 TUESDAY     INTRO TO EARTHSCI.
DAY 3   27 AUGUST 1980 WEDNESDAY
DAY 4   28 AUGUST 1980 THURSDAY
DAY 5   29 AUGUST 1980 FRIDAY
```
1a. Result of E(1)GETTING STARTED and E(2)INTRO TO EARTHSCI.

```
DAY 1   25 AUGUST 1980 MONDAY      GETTING STARTED
DAY 2   26 AUGUST 1980 TUESDAY
DAY 3   27 AUGUST 1980 WEDNESDAY   INTRO TO EARTHSCI.
DAY 4   28 AUGUST 1980 THURSDAY
DAY 5   29 AUGUST 1980 FRIDAY
```
1b. Result of using O(1,1)

```
DAY 1   25 AUGUST 1980 MONDAY      GETTING STARTED
DAY 2   26 AUGUST 1980 TUESDAY     PRETEST
DAY 3   27 AUGUST 1980 WEDNESDAY   INTRO TO EARTHSCI.
DAY 4   28 AUGUST 1980 THURSDAY
DAY 5   29 AUGUST 1980 FRIDAY
```
1c. Result of using E(2)PRETEST

```
DAY 1   25 AUGUST 1980 MONDAY      GETTING STARTED
DAY 2   26 AUGUST 1980 TUESDAY     PRETEST
DAY 3   27 AUGUST 1980 WEDNESDAY
DAY 4   28 AUGUST 1980 THURSDAY
DAY 5   29 AUGUST 1980 FRIDAY      INTRO TO EARTHSCI.
```
1d. Result of using O(2,2)

```
DAY 1   25 AUGUST 1980 MONDAY      GETTING STARTED
DAY 2   26 AUGUST 1980 TUESDAY     PRETEST
DAY 3   27 AUGUST 1980 WEDNESDAY   SCIENTIFIC NOTATION
DAY 4   28 AUGUST 1980 THURSDAY
DAY 5   29 AUGUST 1980 FRIDAY      INTRO TO EARTHSCI.
```
1e. Result of using E(3)SCIENTIFIC NOTATION

```
DAY 1   25 AUGUST 1980 MONDAY      GETTING STARTED
DAY 2   26 AUGUST 1980 TUESDAY     PRETEST
DAY 3   27 AUGUST 1980 WEDNESDAY   SCIENTIFIC NOTATION
DAY 4   28 AUGUST 1980 THURSDAY    INTRO TO EARTHSCI.
DAY 5   29 AUGUST 1980 FRIDAY
```
1f. Result of using C(4)

Fig. 1. Printout resulting from N, E, O and C commands. All samples obtained using P(1,5).

display days 120 through 129. This will not work within the lister (there is no L command in the lister's vocabulary).

Using the Program

A typical work session might go something like this. To start a new lesson plan for earth science, type N to retrieve the blank calendar from the disk. Day 1 will be devoted to handing out books, taking roll, and similar getting-started stuff, so type E(1)GETTING STARTED and hit the return key. For day 2 you might enter E(2)INTRO TO E. SCIENCE. By typing L you get a display of days 1 through 10—days 3 through 10 are still blank (see Fig. 1a). Suppose you want to give a pretest on earth science concepts and it needs to be given on day 2. Type O(1,1) and the introductory activity is moved down to day 3, leaving a blank at day 2 (see Fig. 1b). By typing E(2)PRETEST, the pretest activity is placed at day 2. Listing the first page with the L command confirms it (see Fig. 1c).

After entering other data and rearranging it as needed (see Figs. 1d–1f), you are ready to store the plan on the disk. Typing S causes a request for a file name; enter an appropriate name according to your system notation—I used SYL:EARTHSCI.DAT. If at a future date more changes or additions are necessary, you can retrieve a file by typing R; the computer again will ask for a file name. Be careful with both of these commands and the N command—they will erase all data in memory in the process of loading the new file. However, the most you stand to lose is that day's work—the previous version is still safe on the disk.

If you inadvertently enter a nonexistent file name, the program will stop with an error message; restart the program by entering RUN.

The only time the program requests a file name is when loading an existing file from the disk or when storing a brand new file. It will not ask for a file name when storing an existing file. This eliminates the possibility of wiping out an entire file by entering the wrong file name, or of storing the same file under slightly different file names created by a typographical error.

You can end your session by typing P(1,10) to get a printed copy of the lesson plans. When completed, the computer instructs you to remove the paper, then returns to the main command loop. Stop the program by typing CONTROL-C.

About the Program

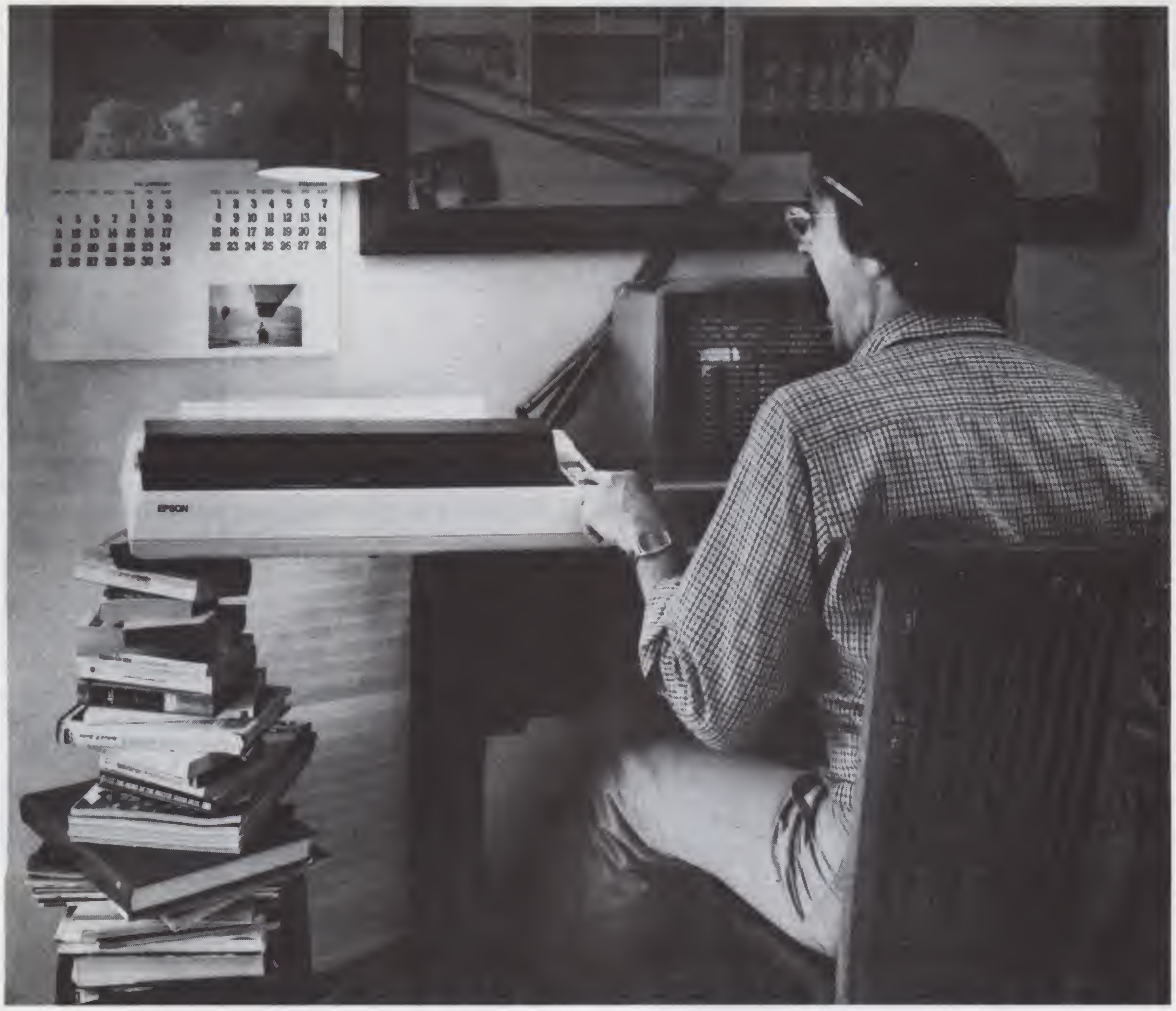
The program itself consists of the main command loop and nine subroutines. The main command loop consists of a page of directions, LINE INPUT statement to allow command entry and a series of IF...THEN statements to decipher the first letter of the command and send control to the correct subroutine. The IF...THEN statement without the GOSUB at line 310 is necessary because the subroutine at line 2570 begins with a CLEAR statement; BASIC would lose track of where the subroutine was called from. At the end of the IF...THEN statements there is an error trap to prevent incorrect entries from occurring.

The Assignment Entry routine's first task is to decipher the rest of the command. Lines 410 through 450 decipher the day number and convert it from string to number format. P$ variable is then assigned the data portion of the original command. The op-

Listing 1. The PLANS.BAS lesson-planning program.

```
100 REM *** PLANS.BAS ***
110 REM JAMES A. INGRAM  804 NO. C ST.
120 REM BROKEN BOW, NEBRASKA     68822
130 CLEAR 12000
140 REM **** MAIN COMMAND LOOP ****
150 PRINT CHR$(12)
160 PRINT "ENTER COMMANDS AS FOLLOWS:"
170 PRINT "     E > ENTER DATA IN BLANK OR CHANGE TO NEW DATA"
180 PRINT "     C > CLOSE GAP IN DATA"
190 PRINT "     O > OPEN GAP IN DATA"
200 PRINT "     H > HELP (PRINT THIS LIST)"
210 PRINT "     S > STORE DATA"
220 PRINT "     R > RETRIEVE DATA"
230 PRINT "     N > BEGIN A NEW FILE"
240 PRINT "     L > LIST DATA ON SCREEN FOR REVIEW"
250 PRINT "     P > PRINT DATA"
260 LINE INPUT "[";C$
270 IF LEFT$(C$,1)="E" THEN GOSUB 390:GOTO 370
280 IF LEFT$(C$,1)="C" THEN GOSUB 500:GOTO 370
290 IF LEFT$(C$,1)="O" THEN GOSUB 660:GOTO 370
300 IF LEFT$(C$,1)="S" THEN GOSUB 860:GOTO 370
310 IF LEFT$(C$,1)="N" THEN 2570: REM CAUSES 'CLEAR' - NO GOSUBS
320 IF LEFT$(C$,1)="R" THEN GOSUB 970:GOTO 370
330 IF LEFT$(C$,1)="L" THEN GOSUB 1330:GOTO 370
340 IF LEFT$(C$,1)="P" THEN GOSUB 1090:GOTO 370
350 IF LEFT$(C$,1)="H" THEN GOSUB 2180:GOTO 370
360 PRINT "INCORRECT ENTRY . . . TRY AGAIN"
370 GOTO 260
380 REM **** ASSIGNMENT ENTRY ROUTINE ****
390 L $=""
400 L=0
410 FOR X=3 TO 5
420 IF MID$(C$,X,1)=")" THEN 450
430 L$=L$+MID$(C$,X,1)
440 NEXT X
450 L=VAL(L$)
460 L0=LEN(C$)-X
470 P$(L)=RIGHT$(C$,L0)
480 RETURN
490 REM **** CLOSE GAP ROUTINE ****
500 L1=0:L1$=""
510 FOR X=3 TO 10
520 IF MID$(C$,X,1)=")" THEN 550
530 L1$=L1$+MID$(C$,X,1)
540 NEXT X
550 L1=VAL(L1$)
560 L=0
570 FOR X=L1 TO L1+10
580 IF LEFT$(P$(X),5)="          " THEN L=L+1 ELSE 600
590 NEXT X
600 FOR Y=L1+L TO P
610 SWAP P$(Y),P$(Y-L)
620 NEXT Y
630 P=P-L
640 RETURN
650 REM ********** OPEN FOR INSERTION ****
660 L1=0:L2=0:L1$="":L2$=""
670 FOR X=3 TO 10
680 IF MID$(C$,X,1)="," THEN 710
690 L1$=L1$+MID$(C$,X,1)
700 NEXT X
710 FOR Y=X+1 TO 10
720 IF MID$(C$,Y,1)=")" THEN 750
730 L2$=L2$+MID$(C$,Y,1)
740 NEXT Y
750 L1=VAL(L1$)
760 L2=VAL(L2$)
770 FOR X=P TO L1+1 STEP -1
780 SWAP P$(X),P$(X+L2)
790 NEXT X
```

More



The MX-100. Not just better. Bigger.

Epson.

Our MX-80 was a pretty tough act to follow. I mean, how do you top the best-selling printer in the world?

Frankly, it wasn't easy. But the results of all our sleepless nights will knock your socks off.

The MX-100 is a printer that must be seen to be believed. For starters, we built in unmatched correspondence quality printing, and an ultra-high resolution bit image graphics capability. Then we added the ability to print up to 233 columns of information on 15″ wide paper to give you the most incredible spread sheets you're ever likely to see. Finally, we topped it all off with *both* a satin-smooth friction feed platen *and* fully adjustable, removable tractors. And the list of standard features goes on and on and on.

Needless to say, the specs on this machine — and especially at under $1000 — are practically unbelievable. But there's something about the MX-100 that goes far beyond just the specs; something about the way it all comes together, the attention to detail, the fit, the feel. Mere words fail us. But when you see an MX-100, you'll know what we mean.

All in all, the MX-100 is the most remarkable printer we've ever built. Which creates rather a large problem for those of us at Epson.

How are we going to top this?

Your next printer.

EPSON

EPSON AMERICA, INC.

116

23844 Hawthorne Boulevard • Torrance, CA 90505 • (213) 378-2220

See the whole incredible Epson MX Series of printers at your Authorized Epson Dealer.

Listing 1 continued.

```
800 FOR X=L1+1 TO L1+L2
810 P$(X)="
820 NEXT X
830 P=P+L2
840 RETURN
850 REM **** STORE ON DISK ****
860 IF F$="" THEN GOSUB 940
870 OPEN "O",1,F$
880 FOR X=1 TO P
890 PRINT #1,A$(X)
900 PRINT #1,P$(X)
910 NEXT X
920 CLOSE #1
930 RETURN
940 LINE INPUT "ENTER FILE NAME : ";F$
950 RETURN
960 REM **** RETRIEVE FROM DISK ****
970 LINE INPUT "ENTER FILE NAME : ";F$
980 OPEN "I",1,F$
990 DIM A$(300),P$(300)
1000 FOR X=1 TO 300
1010 IF EOF(1) THEN 1050
1020 LINE INPUT #1,A$(X)
1030 LINE INPUT #1,P$(X)
1040 NEXT X
1050 P=X-1
1060 CLOSE #1
1070 RETURN
1080 REM **** PRINT HARDCOPY ****
1090 PRINT "TURN ON LINE PRINTER.  HIT RETURN WHEN READY."
1100 LINE INPUT "READY? ";X$
1110 PRINT CHR$(17)CHR$(2)CHR$(30)
1120 L1=0:L2=0:L1$="":L2$=""
1130 FOR X=3 TO 10
1140 IF MID$(C$,X,1)="," THEN 1170
1150 L1$=L1$+MID$(C$,X,1)
1160 NEXT X
1170 FOR Y=X+1 TO 12
1180 IF MID$(C$,Y,1)=")" THEN 1210
1190 L2$=L2$+MID$(C$,Y,1)
1200 NEXT Y
1210 L1=VAL(L1$)
1220 L2=VAL(L2$)
1230 FOR X=L1 TO L2
1240 PRINT "DAY";X;TAB(9)A$(X);TAB(40)P$(X)
1250 FOR Y=1 TO 500:NEXT Y
1260 NEXT X
1270 PRINT CHR$(19)
1280 PRINT CHR$(12)
1290 PRINT "ADVANCE PAPER TO TOP OF NEXT PAGE."
1300 LINE INPUT "HIT RETURN WHEN DONE.  DONE?";X$
1310 RETURN
1320 REM **** SCREEN DISPLAY ROUTINES ****
1330 REM **** LISTING COMMAND LOOP ****
1340 PRINT CHR$(12)
1350 G=LEN(C$)
1355 L$=""
1360 IF G=1 THEN W1=1: GOTO 1430
1370 FOR Y=3 TO 10
1380 IF MID$(C$,Y,1)=")" THEN 1410
1390 L$=L$+MID$(C$,Y,1)
1400 NEXT Y
1410 W1=VAL(L$)
1420 GOTO 1440
1430 PRINT CHR$(12)
1440 FOR X=W1 TO W1+9
1450 PRINT "DAY";X;" ";A$(X);TAB(40)P$(X)
1460 NEXT X
1470 REM
1480 X$=INPUT$(1)
1490 IF X$="F" THEN 1590
1500 IF X$="R" THEN 1690
1510 IF X$="T" THEN 1780
1520 IF X$="B" THEN 1860
1530 IF X$="E" THEN 2150
1540 IF X$="U" THEN 2050
1550 IF X$="D" THEN 1950
1560 PRINT "INCORRECT INPUT -- TRY AGAIN . . ."
1570 GOTO 1430
1580 REM **** FORWARD ONE PAGE ****
1590 W1=W1+10
1600 IF W1>300 THEN W1=300
1610 W2=W1+9
1620 IF W2>300 THEN W2=300
1630 PRINT CHR$(12)
1640 FOR X=W1 TO W2
1650 PRINT "DAY";X;" ";A$(X);TAB(40)P$(X)
1660 NEXT X
1670 GOTO 1470
1680 REM *** REVERSE ONE PAGE ***
1690 PRINT CHR$(12)
1700 W1=W1-10
1710 IF W1<1 THEN W1=1
1720 W2=W1+9
1730 FOR X=W1 TO W2
```

More

eration of the MID$ function is as follows: C$ is the string variable being considered, X is the first character to be pulled, and 1 is the number of characters to be pulled. As each consecutive character is pulled, it is added to the string variable L$ until the limit is reached. L0 is the length of the data portion of the command, needed in the RIGHT$ statement to separate the data from the initial command characters. L is the subscript of P$, which is the same as the day number.

The Close Gap and Open Gap subroutines operate in about the same way. Lines 510 through 650 and 670 through 760 decipher the commands. The remaining lines move data up or down an appropriate number of lines in the P$ array. Lines 570 through 590 determine the number of lines in the gap to close.

A word of caution here: the use of the Microsoft SWAP command is mandatory on the H-8 version of BASIC. I started with a standard reassignment of subscripts, such as P$(X) = P$(X + L2), and had internal time delays of up to two minutes for the insertion or deletion of a single line. A call to Microsoft provided the explanation. The above equality is performed by copying the data into new string space, assigning the new variable name, erasing the old string (garbage collecting in computerists' words), and compressing the remaining string space. Needless to say, with 12K of string space, this takes time. On the other hand, the SWAP P$(X),P$(X + L2) command is a true swap; it uses no extra string space, no garbage collecting, no condensing and no wasted time. The two minute delay is reduced to two seconds. You will probably have to experiment with your own computer to see how it handles this situation.

The Store and Retrieve subroutines are simple. They either print or input data from the disk in two variables; A$ for the day/date and P$ for the assignment. The data is numbered as it is read in; day numbers are not stored on the disk. Microsoft syntax opens files with the OPEN "I",1, "FILENAME" which translates to OPEN CHANNEL #1 FOR INPUTTING THE FILE "FILENAME". Output changes the I to O. Files are closed with the CLOSE #1 statement.

The Print Hardcopy routine contains the familiar command translation routines in lines 1130 through

1220. Lines 1230 through 1260 print the requested lesson plans. Line 1250 provides a time delay to allow the printer to keep up without losing whole chunks of data. The variable X$ is just a *junk* variable, allowing use of the return key for restarting execution after a pause.

Some quirks of my system become evident here. The printer is hooked up to the terminal, not to a separate port; whatever appears on the screen also appears on the printer if turned on. Operation of the printer is controlled through the CHR$ function. Here is a summary of such commands found in the program:

- CHR$(2)—normal width characters (not enhanced)
- CHR$(12)—screen erase
- CHR$(17)—turn printer on
- CHR$(19)—turn printer off (leaves paper advance on)
- CHR$(30)—12 characters per inch

The screen display routines (Lister) are all basically similar. Each defines a window whose first line is W1 and which is 10 lines long; W2 represents the last line of the window. P represents the last line of recorded activities. Even if activities are pushed off the bottom end of the calendar of class days, they are not lost. The storage routine also makes use of this variable. In this way my disk file may grow beyond its original size of 50 sectors or so. Provision is made for 300 entries. Each routine has appropriate IF. . . THEN checks in the case of negative lines or lines higher than 300 being requested.

The command summary routine, Help, is largely self-explanatory. Again, X$ is a junk variable, and LINE INPUT was used only to eliminate Microsoft's prompt and the waste of a line at the bottom of the screen.

The last routine, Start New File, is the same as the Retrieve routine with two exceptions. It reads only the SY1: CLASDAYS.DAT file, and produces rather than reads the strings of 39 blanks found under P$. Note that it destroys all variable values and subroutine calls, since it starts with the CLEAR statement. Line 2610 senses when the last line of the file has been input. This is a nice feature of Microsoft BASIC which eliminates the need to know the number of items in a file before it is read.

However your computer operates, I hope you enjoy using and improving on the ideas presented here.■

Listing 1 continued.

```
1740 PRINT "DAY";X;" ";A$(X);TAB(40)P$(X)
1750 NEXT X
1760 GOTO 1470
1770 REM *** FIRST PAGE ***
1780 PRINT CHR$(12)
1790 W1=1
1800 W2=W1+9
1810 FOR X=W1 TO W2
1820 PRINT "DAY";X;" ";A$(X);TAB(40)P$(X)
1830 NEXT X
1840 GOTO 1470
1850 REM *** LAST PAGE ***
1860 PRINT CHR$(12)
1870 W2=P
1880 W1=P-9
1890 IF W1<1 THEN W1=1
1900 FOR X=W1 TO W2
1910 PRINT "DAY";X;" ";A$(X);TAB(40)P$(X)
1920 NEXT X
1930 GOTO 1470
1940 REM *** UP ONE LINE ***
1950 PRINT CHR$(12)
1960 W1=W1+1
1970 W2=W1+9
1980 IF W2>300 THEN W2=300
1990 IF W1>300 THEN W1=300
2000 FOR X=W1 TO W2
2010 PRINT "DAY";X;" ";A$(X);TAB(40)P$(X)
2020 NEXT X
2030 GOTO 1470
2040 REM *** DOWN ONE LINE ***
2050 PRINT CHR$(12)
2060 W1=W1-1
2070 IF W1<1 THEN W1=1
2080 W2=W1+9
2090 IF W2>300 THEN W2=300
2100 FOR X=W1 TO W2
2110 PRINT "DAY";X;" ";A$(X)TAB(40)P$(X)
2120 NEXT X
2130 GOTO 1470
2140 REM *** EXIT WITHOUT SCREEN ERASE  ***
2150 PRINT "NOW IN MAIN COMMAND MODE.  ENTER COMMAND:"
2160 GOTO 260
2170 REM **** COMMAND SUMMARY ****
2180 PRINT CHR$(12)
2190 PRINT "HIT RETURN TO ADVANCE TO NEXT PAGE OF INSTRUCTIONS."
2200 PRINT
2210 PRINT "   E(#)DATA-DATA-DATA"
2220 PRINT "     ENTERS DATA AT DAY NUMBER # UP TO 39 CHARACTERS LONG."
2230 PRINT
2240 PRINT "   C(#)"
2250 PRINT "    CLOSES GAP OF BLANK DATA UP TO 10 LINES BY MOVING FOLLOWING"
2260 PRINT "    DATA UP ON CALENDAR."
2270 LINE  INPUT  "";X$
2280 PRINT CHR$(12)
2290 PRINT "   O(#1,#2)"
2300 PRINT "     OPENS GAP FOR INSERTION OF DATA BEGINNING AFTER LINE #1"
2310 PRINT "     FOR A SPACE OF #2 LINES BY MOVING DATA DOWN ON CALENDAR."
2320 PRINT "     DATA BEYOND END OF CALENDAR IS RETAINED UP TO 300 ENTRIES."
2330 PRINT
2340 PRINT "   H"
2350 PRINT "     HELP -- PRINT THESE INSTRUCTIONS."
2360 PRINT
2370 PRINT "   S"
2380 PRINT "     STORE CURRENT DATA ON DISK.  USES INPUT FILE NAME IF"
2390 PRINT "     AVAILABLE, ASKS FOR FILE NAME IF NOT AVAILABLE."
2400 LINE INPUT  "";X$
2410 PRINT CHR$(12)
2420 PRINT "   N"
2430 PRINT "     BEGINS NEW FILE BY FURNISHING CALENDAR WITH ASSIGNMENT"
2440 PRINT "     DESCRIPTIONS BLANK."
2450 PRINT
2460 PRINT "   L"
2470 PRINT "     LISTS DATA ON SCREEN IN 10-LINE PAGES.   DATA MAY BE"
2480 PRINT "     SCROLLED UP OR DOWN A LINE OR A PAGE AT A TIME.  SPECIFIC"
2490 PRINT "     DIRECTIONS GIVEN WITHIN ROUTINE."
2500 PRINT
2510 PRINT "   P(#1,#2)"
2520 PRINT "     PRINTS DATA ON LINE PRINTER FROM LINE NUMBER #1 TO #2."
2530 LINE INPUT "";X$
2540 PRINT CHR$(12)
2550 RETURN
2560 REM *** START A NEW FILE ***
2570 CLEAR 10000
2580 DIM A$(300),P$(300)
2590 OPEN "I" ,1,"SY1:CLASDAYS.DAT"
2600 FOR X=1 TO 225
2610 IF EOF(1) THEN 2650
2620 INPUT #1,A$(X)
2630 P$(X)="
2640 NEXT X
2650 CLOSE #1
2660 P=X-1
2670 GOTO 260
```

THE COPYRIGHT KIT

How to Copyright Your Computer Software

A self-instruction booklet on copyrighting the computer programs you write. IF YOU EVER WROTE A COMPUTER PROGRAM YOU NEED THIS BOOK! INCLUDES: Step by Step instructions, sample forms, as well as discussions of copyrights, patents and trade secrets, your rights secured by copyright, legal remedies upon infringements, material not copyrightable and MUCH MORE! Written by Attorneys. Published by National Attorneys' Publications, Inc. and distributed exclusively through B.T. Enterprises.

ONLY $11.95

BT ENTERPRISES

B.T. Enterprises
171 Hawkins Road
Centereach, New York
(516) 981-8568 (Voice)
(516) 588-5836 (Data)
MNET-70331, 105

VISA — master charge

124 Add $2.00 S & H, NYS res. add appr. tax

TEXAS COMPUTER SYSTEMS

Offers Lowest Prices on

TRS-80

COMPUTERS

Model II 64K $3349 — Free with purchase: 1 Box of 10 Diskettes a $69 Value		Epson Printers $ Call
Model III 16K $849 Model III 32K 2 Disks $2100	Color Computer 4K Lev. I $329 16K Ex. Basic $489	Radio Shack Stereos, Radios Discounted Call for discount on orders over $100

For fast, efficient service, we can air freight from Dallas to major a/p near you. Call for information

- Payment: Money Order, Cashier's Check, Certified Check. Personal checks take 3 wks. VISA, MC add 3%
- Prices subject to change any time
- No tax out of state. Texans add 5%
- Delivery subject to availability
- Shipping extra, quoted by phone

TEXAS COMPUTER SYSTEMS 328

Box 951, Brady Texas 76825

800-351-1473 800-433-5184

Texas Residents 817-274-5625

---

progressive computing

Z-FORTH IN ROM by Tom Zimmer
5 to 10 times faster than Basic. Once you use it, you'll never go back to basic! $ 75.00
source listing add $ 20.00

OSI FIG-FORTH True fig forth model for OS65D with fig-editor named files, string package & much more $ 45.00

TINY PASCAL Operates in fig-forth, an exceptional value when purchased with forth.
TINY PASCAL & documentation $ 45.00
FORTH & TINY PASCAL $ 65.00

SPACE INVADERS 100% machine code for all systems with 64 chr. video. Full color & sound on C2, 4P & 8P systems. The fastest arcade program available $ 14.95

PROGRAMMABLE CHARACTER GENERATOR $ 99.95
Use OSI's graphics or make a complete set of your own! Easy to use, comes assembled & tested
2 Mhz. boards $109.95

PROGRAMMABLE SOUND BOARD $ 74.95
Complete sound system featuring the AY-3-8910 sound chip.
Bare boards available. $ 29.95

32/64 CHARACTER VIDEO MODIFICATION $ 39.95
Oldest and most popular video mod. True 32 chr. C1P, or 32/64 chr. C4P video display. Also adds many other options.

ROMS!!!
Augment Video Mod with our Roms. Full screen editing, print at selectable scroll, and many more features.
Basic 4 & Monitor $ 49.95
Basic 3 $ 18.95
All 3 for $ 65.00

65D DISASSEMBLY MANUAL. by Software Consultants
First class throughout. A must for any 65D user. $ 24.95

NUMEROUS BASIC PROGRAMS, UTILITY PROGRAMS AND GAMES ALONG WITH HARDWARE PROJECTS. ALL PRICES ARE U.S. FUNDS. Send for our $1.50 catalogue with free program (hardcopy) Memory Map and Auto Load Routine.

CHARGEX VISA

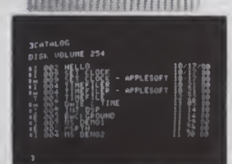
OSI Software & Hardware 202
3336 Avondale Crescent
Windsor, Ontario, Canada N9E 1X6
(519) 969-2500
3281 Countryside Circle
Pontiac Township, Michigan 48057
(313) 373-0468

IT'S ABOUT TIME!

SUPERCLOCK II™

A COMPLETE CLOCK/CALENDAR SYSTEM FOR THE APPLE II.

FEATURES:
- Timing from milliseconds to 99 years
- 12/24 Hour formats plus day of week
- Does not use C800-CFFF address space
- Automatic dating of files stored on disk
- Automatic updating of PASCAL's Filer
- Up to four software controlled interrupts
- Full battery operation for up to 10 years



SUPERCLOCK II COMPLETE SYSTEM **$159**

NEW! **TIME-CLOCK II** program automatically keeps track of the time you spend on your computer for each job, client, program, etc. Then prints out a detailed report. Requires SUPERCLOCK II, Applesoft, and disk **$30**

west side electronics

P.O. Box 636D, Chatsworth, CA 91311 Phone (213) 884-4794

All orders - add $3.50 for postage, insurance, and handling ($7.00 outside Continental USA). California residents add 6% sales tax. A 3% surcharge will be added to all credit card orders.

Apple, Apple II, and Applesoft are trademarks of Apple Computer, Inc.

ADVENTURE INTERNATIONAL

NEW

THE AdveNture Continues

The Scott Adams' Adventure series announces the release of Adventure #11 *"Savage Island,* part 2" and Adventure #12 *"Golden Voyage"*.

For The • TRS-80 • APPLE • ATARI • $19.95 tape

Also available on Bonus Triple Pack Disk Adventures #10, #11, and #12.
For the TRS-80 and Apple $39.95

2 maces & magic SERIES

NEW

STONE of Sisyphus

The Stone of Sisyphus carries you to the "thinking man's" dungeon. A subterranean world of monsters, magic, traps and treasures demands brains rather than luck to survive (of course, you might take along your mace just in case . . .).

MACES & MAGIC are fantasy adventures involving you and your computer Armed only with your wits, a microcomputer, and the software provided, you can become the hero or the meal your destiny dictates. You create a character, equip him (or her) with suitable weapons and armor, and enter the dungeon in search of fame and fortune Neither is particularly easy to obtain

If you are successful in avoiding or conquering the various monsters, traps, enchantments and illusions set by our nefarious dungeonmasters, you may escape with riches and glory Your name and deeds will be recorded for posterity in the records of the dungeon More importantly, you'll be alive. You may then use the same character in his more experienced and wealthy form when you enter dungeons on later occasions

The object of the whole exercise is not just to fight the monsters and collect treasure You have to get out alive to enjoy it In every dungeon there is at least one exit It is possible to escape from each and every dungeon with a whole skin We state that fact here because players often believe this not to be true We really aren't out to get you Not really

FOR ALL OF THE BELOW
Apple 2 Disk with Applesoft
or Apple 2 Plus 48K
TRS-80 32K Model 1 or
TRS-80 48K Model 3

032-0100
$29.95

TWO FULL DISKS WORTH OF DATA!!

A compacted data structure makes it possible to run this fantasy on a single disk drive, though 2 full diskettes worth of data are supplied. Expanded use of free-form input puts your ingenuity to an even greater test. The responsiveness of the programs to the individual qualities of the character make this adventure frustratingly enjoyable for hundreds of hours before its secrets can be unlocked.

SCOTT ADAMS SEAL OF APPROVAL

A Division of Scott Adams, Inc.
Adventure International • Box 3435 • Longwood, FL 32750
TOLL-FREE ORDER NUMBER: (800) 327-7172
IN FLORIDA (305) 862-6917
SHIPPING AND HANDLING CHARGES ARE EXTRA
PRICES ARE SUBJECT TO CHANGE WITHOUT NOTICE

✓109

ai

CompuServe is a lot more than fun and games and news.

Although we're probably best known for our consumer-oriented services, CompuServe maintains powerful services for the serious computer user. CompuServe has helped some of the nation's largest companies and financial institutions through a wide range of business and scientific-oriented computer programs. With our Information Service, we can extend this expertise to you. Here are some of the programs available to the personal computer user through the CompuServe Information Service:

PASCAL: DEC User Group version of this famous language.
XF4: Extended CompuServe Fortran compiler.
MACRO: Advanced Macro Assembly language for PDP-10. Talk about fast.
FILGE: Fast, easy file generator and editor. Line oriented, no line numbers required!
XBASIC: CompuServe extended BASIC.
JUMBLE: File encryption program.
TECO: High level text editor.
LINK: Linking loader program.
DDT: Dynamic debugging tool.
CREF: Symbolic cross-reference generator used with language compilers.
FILCOM: File comparison program.
SALARY: Calculates salary and increases in various time frames including per year, month, period and hour. This program is intended primarily for managers doing reviews.
SNOBOL: String manipulation language.
RUNF10: Word processing and text formatting program (write resumes etc.).
CONCOR: Concordance generator. This program is very useful for documentation purposes. CONCOR reads an ASCII file and creates an output file which contains a line numbered listing of the original file, and a list of all the words contained in the file along with the numbers of the lines on which each word occurs (similar to an index).
BLIS10: High level implementation language for DEC PDP-10.
BINED: Binary file editor.
APL: A Programming Language.
FINTOL: Financial analysis tool. Contains several program modules which can perform financial analysis for business or personal needs.
—Loan Payment and Amortization
—Compound Interest Calculations
—Sum from a Periodic Investment
—Sinking Fund Deposit
—Present Value of Cash Flows
—Present Value and Internal Rate of Return (Capital Projects)
—Depreciation Analysis
—Compound Growth Rates
RANNO: Random number generator.
XEDIT: XF4 binary file editor.

We also offer high speed professional line printer service, error-free file transfer, CompuServe's Software Exchange, and computer manufacturers' newsletters. Computer Users Groups can exchange information and up-to-the-minute gossip in the computer world.

And . . . if Aunt Matilda comes over, you can still let her play Adventure. Demonstrations and software at all Radio Shack® Computer Centers and many Radio Shack® outlets.

CompuServe 147

Information Service Division
5000 Arlington Centre Boulevard
Columbus, Ohio 43220
(614) 457-8600

Will the wave of Japanese micros into the U.S. knock the socks off American firms? The first installment of this series examining these Japanese imports spotlights the Casio FX-9000P.

Japanese Invasion: Part 1

By G. Michael Vose

Japanese technology and productivity are being discussed at a lot of business conferences in our country. Japanese cars, Japanese television sets and other Japanese products have many American manufacturers looking over their collective shoulder. Japanese entrepreneurship has produced many innovative products and *very* competitive prices.

And so it is with microcomputers. The long-awaited entry of the Japanese into the meteoric small-computer market has begun. In the last six months, Sharp, NEC, Canon, Casio and Toshiba have introduced microcomputers to their domestic markets and to the world at large. They have penetrated the European market and are seeking a toehold in the United States as well. Most of these companies already have marketing organizations in place to sell such products as watches, cameras, stereos and calculators.

During the next few months, *Microcomputing* will take a close look at what the Japanese have to offer to the buyer of small-computer products. Specifically, we'll examine and review new microcomputers made by NEC, Sharp, Toshiba and Canon and, beginning with this installment, Casio, Inc.

The Casio FX-9000P

Casio has grown substantially over the last several years, due largely to their success in the calculator and digital watch markets. They produce a full line of calculators, from small, credit-card-sized models to fully programmable scientific and engineering calculators. The microcomputer is a natural extension of this line. Casio has responded with a hand-held pocket computer called the FX-702P, and with a desktop computer called the FX-9000P.

The FX-9000P is a compact computer. In size and design, it most closely resembles Hewlett-Packard's new HP-85 (see review on pg. 120). It offers expandable RAM up to 32K and a "Z-80 compatible processor."

The FX-9000P uses Casio's CA-BASIC, a proprietary BASIC using a ROM chip BASIC interpreter. CA-BASIC has some excellent graphics capabilities.

The FX-9000P is so new that Casio has not yet produced disk drives for the unit, but there is a port for mini-floppy disk drives (which are under development), as well as a cassette tape I/O port. The unit uses any standard cassette tape recorder. The computer comes with an Epson MX-82 graphics printer.

The cream-colored cabinet contains the 5-1/2 inch diagonal measure, 16×32 column video display screen and a 62-key keyboard. To the right of the screen is a small door. Inside this door are four slots, which accommodate plug-in RAM and ROM modules. These modules are available as 4K ROMs, 4K RAMs and 16K RAMs. The only currently available ROM pack, called the E-4K ROM module, is used to permit matrix operations.

A unique feature of the FX-9000P is its battery-backed RAM storage. The C-4K RAM modules can store data or programs even when the power to the computer is turned off. Since cassette tape is the only storage medium currently available for the FX-9000P, this feature could prove to be useful for storing a few often-used programs. RAM modules are far more expensive than cassette tape, however, so this feature has a practical limit.

An Electronic Slide Rule

As a logical extension to their line of both programmable and nonprogrammable calculators, the FX-9000P

The Casio FX-9000P.

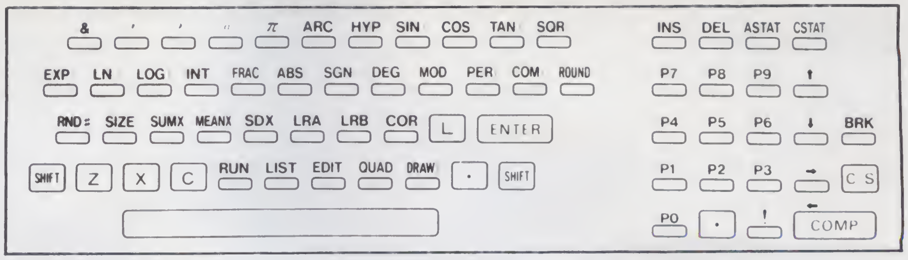
The open cover reveals four slots for plug-in RAM and ROM modules.

Table 1. Casio's mathematic function keys (from the Casio FX-9000P operation manual).

Function name		Format	
Trigonometric function	$\sin x$	SIN(x)	[SHIFT SIN(]
	$\cos x$	COS(x)	[SHIFT COS(]
	$\tan x$	TAN(x)	[SHIFT TAN(]
Inverse trigonometric function	$\sin^{-1} x$	ASN(x)	[SHIFT ARC SHIFT SIN(]
	$\cos^{-1} x$	ACS(x)	[SHIFT ARC SHIFT COS(]
	$\tan^{-1} x$	ATN(x)	[SHIFT ARC SHIFT TAN(]
Hyperbolic function	$\sinh x$	HSN(x)	[SHIFT HYP SHIFT SIN(]
	$\cosh x$	HCS(x)	[SHIFT HYP SHIFT COS(]
	$\tanh x$	HTN(x)	[SHIFT HYP SHIFT TAN(]
Inverse hyperbolic function	$\sinh^{-1} x$	AHS(x)	[SHIFT ARC SHIFT HYP SHIFT SIN(]
	$\cosh^{-1} x$	AHC(x)	[SHIFT ARC SHIFT HYP SHIFT COS(]
	$\tanh^{-1} x$	AHT(x)	[SHIFT ARC SHIFT HYP SHIFT TAN(]
Square root	$\sqrt{x}$	SQR(x)	[SHIFT SQR(]
Exponential function	e^x	EXP(x)	[SHIFT EXP]
Natural logarithm	$\ln x$	LN (x)	[SHIFT LN]
Common logarithm	$\log x$	LOG(x)	[SHIFT LOG]
Factorial	$x!$	x !	[SHIFT !]
Conversion into integer	INT x	INT(x)	[SHIFT INT]
Memory size	SIZE		[SHIFT SIZE]
Random number	RND#		[SHIFT RND#]
Statistics calculation			
Number of data items n	CNT		—
Standard deviation of x	SDX		[SHIFT SDX]

has a multitude of mathematic function keys (see Table 1). These functions are engaged when the keyboard is in the uppercase (shift-lock) mode. The variety of calculations that the machine can perform, either in its direct or program mode, seems to indicate that Casio is courting the scientific and engineering applications market. In this realm, the FX-9000P is superb. Many of the mathematic functions can be plotted graphically, as well. This combination of features will put a substantial number of these machines in laboratories and research facilities around the world.

In addition to the pure mathematic functions, the FX-9000P will also perform a variety of statistical functions.



Key functions in the shift-in mode (reprinted from the operation manual).

It will perform standard deviation and linear regression calculations automatically. It has function keys to perform permutation and combinatorial calculations. There is even a function key for the conversion of sexagesimal numbers to decimal numbers. The operating range of the FX-9000P is from $+1 \times 10^{-99}$ to $+9.999999999999 \times 10^{99}$.

Although the FX-9000P has a small screen, it uses high-resolution graphics. The system uses several very flexible graphics commands which permit the production of sophisticated graphs and figures. This capability is available in either the direct (command) mode or the program mode. These graphics can be output to the Epson MX-82 printer for high quality hard copy.

	$x\sigma_{n-1}$		
Standard deviation of y	$y\sigma_{n-1}$	SDY	——
Average of x	$\bar{x}$	MEANX	[SHIFT][MEANX]
Average of y	$\bar{y}$	MEANY	——
Sum of x	Σx	SUMX	[SHIFT][SUMX]
Sum of y	Σy	SUMY	——
Square sum of x	Σx^2	SUMX2	——
Square sum of y	Σy^2	SUMY2	——
Data product sum	Σxy	SUMXY	——
Constant term	A	LRA	[SHIFT][LRA]
Regression coefficient	B	LRB	[SHIFT][LRB]
Correlation coefficient	r	COR	[SHIFT][COR]
Removal of integer part	FRAC x	FRAC(x)	[SHIFT][FRAC]
Conversion into absolute value	$\lvert x \rvert$	ABS(x)	[SHIFT][ABS]
Positive/negative sign giving	Positive number → 1 0 → 0 Negative number → -1	SGN(x)	[SHIFT][SGN]
Degree, minute, second (sexagesimal number → decimal number)		DEG(x)	[SHIFT][DEG]
Remainder calculation		MOD(x, y)	[SHIFT][MOD]
Permutation	$_xP_y$	PER(x, y)	[SHIFT][PER]
Combination	$_xC_y$	COM(x, y)	[SHIFT][COM]
Rounding		ROUND(x, y)	[SHIFT][ROUND]

(The y-th significant digit of x is *counted* as a unit when it is 5 or over, or disregarded when it is below 5.)

CA-BASIC

Casio has developed its own version of BASIC for the FX-9000P. It has several nice features and some inadequacies, as well.

Table 2 shows the commands supported by CA-BASIC. KEYIN is equivalent to Microsoft BASIC's INKEY$ command, and the ROPEN, RPUT and RGET statements are used for writing and retrieving data to and from cassette tape. There are four screen output commands; TAB and CSR direct output to specific areas of the video display and REV and NORM produce normal (white on black) or inverse (black on white) video. The screen is actually green in color and is surprisingly easy on the eyes, given its small size.

CA-BASIC also supports CHR$, MID$ and STR$ functions. It uses standard ASCII character codes and adds several graphics functions to the graphics commands. DOT, GIN$ and GOUT, CHGX, CHGY and POS are display output formatting functions.

The FX-9000P uses a line editor for editing. There are four cursor control keys for moving the cursor; two of these keys double as insert and delete keys in the shift-lock mode. The editor lets you modify entire program lines, including line numbers. This feature provides a method of renumbering individual program lines or entire programs (the latter process is a bit tedious). The editor remains en-

Direct Commands:	PRG, LIST, EDIT, RUN, CLEAR, CLEAR A, PASS, RFILE, RLIST, RCLEAR, RSAVE, RLOAD.
Basic Commands:	REM, LET, READ, DATA, RESTORE, INPUT, KEYIN, PRINT, IF-THEN, GoTo, GoSUB, RETURN, ON-GoTo, ON-GoSUB, FOR, NEXT, SET, STOP, END, DIM, CLEAR DATA, SAC, STAT, ROPEN, RPUT, RGET, RCLEAR.
Graphics Commands:	CLEAR DISP, INIT, DRAW, CDRAW, QUAD, CQUAD.
Numerical Functions:	SIN, COS, TAN, ASN, ACS, ATN, HSN, HCS, HTN, AHS, AHC, AHT, SQR, EXP, LN, LOG, INT, FRAC, ABS, SGN, DEG, MOD, PER, COM, ROUND, RND#, SIZE, CNT, SUMX, SUMY, SUMXY, SUMX2, SUMY2, MEANX, MEANY, SDX, SDY, LRA, LRB, COR, VAL, LEN, ASC, p.
Character Functions:	MID$, CHR$, STR$.
Output Control Functions:	TAB, CSR, REV, NORM.
Graphics Functions:	DOT, CHGX, CHGY, GIN$, GOUT, POS.

Table 2. Commands supported by CA-BASIC (from the operation manual).

gaged until you press the break key to signify that you are through editing; when one line is edited, the editor falls through to the next line and awaits your changes.

The FX-9000P limits variables to 251 (A, B . . . A1, A2 . . . Z9) and imposes a limit of 94 characters per line of BASIC code. Arithmetic hierarchy is the same as Microsoft BASIC and the FX-9000P uses floating point arithmetic. The random number generator uses numbers between 0 and 1, as do versions of Microsoft BASIC, such as Apple Computer's Applesoft.

The FX-9000P has nine separate program areas: P0 to P9. Each of these areas can hold a program independent of the other eight. It is possible, therefore, to have nine separate programs in memory at any given time. Each of these program areas is 2.57K long. This is enough memory for 27 lines of BASIC code, with each line running to the maximum of 94 characters per line. If the programmer limits his code to 32 characters per line, he can write 80 program lines in each of the nine program areas.

CA-BASIC is a little slow. A FOR–NEXT loop counting from 1 to 1500 takes about a minute. On a TRS-80, such a loop would take about 20 seconds.

The Future of the FX-9000P

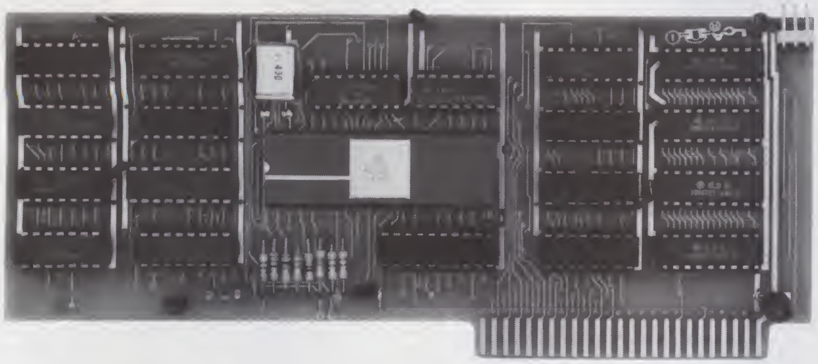
As an engineering and scientific applications computer, the FX-9000P has great potential. The single-key math function commands will aid the user in programming or in calculating from the direct mode. Using the uppercase keyboard mode for math functions does seem to rule out the future incorporation of lowercase capability. This would mean the FX-9000P will not be adaptable for word processing applications (the screen is too small for this use anyway).

The FX-9000P is priced at $720 for the base unit, with the MX-82 printer adding roughly $500. With this kind of price tag, the machine will be very competitive with American-made computers. The FX-9000P should be attractive to industrial and engineering buyers—maybe for these groups, at least, the era of a computer on every desk will become a reality.

Whether Casio will develop a larger machine for word processing or a machine flexible enough for the office and home environment will depend on how well the FX-9000P can be marketed.

This may depend on software support. Casio has produced little software for the FX-9000P. Engineers and scientists familiar with FORTRAN or BASIC will be able to produce their own software with minimal study. And beginning computerists will want to learn CA-BASIC by writing their own programs. But businessmen and other nontechnical people will want ready-to-run software, and Casio will have to produce it if the FX-9000P is to gain broad acceptance.■

NOW THAT YOU'RE SEEING DOUBLE YOU NEED MORE ROOM...



DOUBLEVISION™

80 x 24 Video Display from Computer Stop

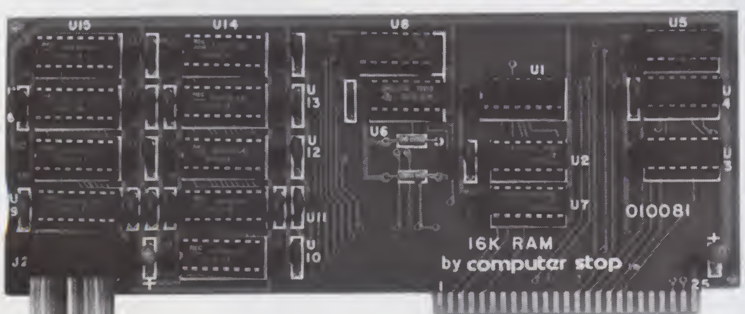
DoubleVision is the time-tested upper-and-lower-case Video Display. And now it has a host of new features. Like the availability of a completely commented source listing of software and hardware schematics. Or the inclusion of a Sysgen program to select type of cursor, monitor band width and display of control characters.

Moreover, DoubleVision is compatible with CCA Data Management (and full screen mapping), Easywriter Professional System, Apple Pie 2.0, PASCAL, Z-80 Softcard, MAGIC-WAND and WORDSTAR.

It's the only 80-column card that has software on disk making it easy to provide updates. In fact, updates are automatically mailed to our customers and *billed* for cost of disk and shipping only. In short, DoubleVision is adaptable to meet your needs.

The price? It's just $295.

...AND HERE'S MORE ROOM!



The CS16K RAM Card

(compatible with PASCAL, CP/M®, DOS 3.3 and other Apple software)

When you add the CS16K RAM Card to your Apple II with 48K bytes of RAM already in place, you immediately increase your capacity to 64K.

The CS16K RAM Card is compatible with PASCAL, Digital Research's CP/M, DOS 3.3, COBOL, FORTRAN, Personal Software's VisiCalc, PILOT, Integer BASIC, Applesoft BASIC and other software currently used with APPLE II.

What's more, the CS16K RAM Card is backed by the Computer Stop reputation for excellent service. In fact, **we've doubled the warranty to 180 days,** instead of the usual 90.

And we've saved the best 'til last. The Price. It's only $195.

computer stop Your best stop for Apple products.

16919 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-8500

Available at select Apple dealers.

283

Apple is a trademark of Apple Computer, Inc.
CP/M is a trademark of Digital Research

2545 W. 237th Street
Torrance, CA 90505
(213) 539-7670

Dealers: Ask about special-priced demo unit.

With the flick of a switch, you can go from Apple DOS 3.2 to 3.3 and back again.

DOS Mod

By Scott King

If you just bought a new Apple DOS 3.3 update kit, you are probably learning what a great joy it can be to convert those old DOS 3.2 disks from 13 to 16 sectors.

If, like me, you have a single disk drive, updating with the Muffin program looks like an endless process of switching disks—pulling this one out, putting that one in—for *every* program on *every* disk you own. It might be easier, after all, to use the BASIC disk and run DOS 3.2 disks on your 3.3 system. But that means using two disks when one should do.

There is another alternative. By making a simple hardware modification, you can have your cake and eat it too (or in this case, your 16 sectors and DOS 3.2). All you need is a single-pole double-throw switch, three pieces of wire (12 inches long), a soldering iron and your old DOS 3.2 ROMs. Soon you'll be able to go from DOS 3.3 to 3.2 and back again with the flick of a switch.

This procedure isn't difficult, but you should be careful. In fact, if you've never worked with ICs, I don't suggest that you try this alone. It's much better to seek good help than to suffer the consequences of inadvertent mistakes.

Modification Steps

1. Locate ROM P5A on the lower left of the disk-controller board.

2. Using the IC tool supplied with the DOS 3.3 Update kit, gently pry out the ROM, making a note of the direction in which it was installed.

3. Carefully set this ROM down and place DOS 3.2 ROM P5 on top of it, making sure that the notch on each is at the same end (see Fig. 1). This is to ensure that pin 1 of P5 is touching pin 1 of P5a. Be careful so that the legs of one are not shorting out adjacent legs of the others (i.e., that pin 3 of the top IC is touching pin 3 of the bottom IC, not pin 2 or 4).

4. When the two ICs are lined up correctly, carefully solder pin 1 of the top IC to pin 1 of the bottom IC. (*Note*: use just enough solder to join them. Do not allow solder to run down to the bottom of the pin or you won't be able to plug the ICs back into the socket afterward.)

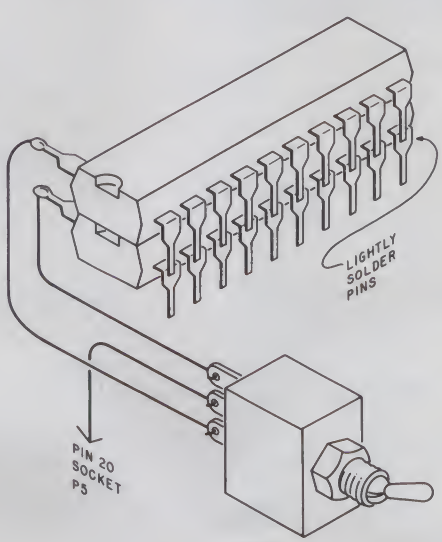


Fig. 1.

5. Continue soldering corresponding pins around the ICs, one at a time. Allow each solder connection to cool down before going on to the next. *Do not solder pins 20*. These will be used for switching.

6. Using needle-nosed pliers carefully bend pin 20 of each IC out straight as shown in Fig. 1. Do not flex these pins back and forth. They are very fragile and may break. Make one smooth bend and stop.

7. Solder one end of one piece of wire to pin 20 of the bottom IC. Solder the other end of the wire to one of the leads of the switch.

8. Solder another piece of wire to pin 20 of the top IC. (If it looks like these two pins might ever come together, wrap them each with electrical tape to keep them apart.) Solder the other end of this wire to the other lead of the switch.

9. Solder one end of the last piece of wire to the center pin of the switch. (It might be labeled C or Comm.)

10. Take the whole assembly back to the disk-controller card and gently reinstall the piggybacked ICs in the socket marked P5. Make sure they are facing in the direction noted back in step 2.

11. Gently turn the card over and locate the lead that would have been

Address correspondence to Scott D. King, 7905 59th Ave. N., New Hope, MN 55428.

used for pin 20 of IC P5. It should be at the lower right of the board.

12. Carefully solder the remaining end of the third wire to this point. Since pin 20 is the power pin for each ROM, the switch will "turn on" either the old or the new ROM.

13. Reinstall the disk-controller card into the Apple, making sure that the wires do not short to any other points and that the switch is not shorting out on anything. Turn on the computer and boot up a DOS 3.3 disk. If the disk will not load, press reset to stop the drive, flick the switch to the other position and try again. This time it should come up. Now that you have DOS 3.3 up, try a 3.2. Stop the computer and insert a DOS 3.2 disk into the drive. Move the switch over to the other position and boot up the disk. The 3.2 disk should load as it used to. Mark which position is which on the switch and find a good place to mount it. (I put mine just above and to the right of the keyboard.)

Now you can boot anything directly without having to worry about which DOS it requires.■

The Ascent Of Computers

By James R. Avoli

To unearth your computer's roots, you have to go back many centuries and across several continents.

Most of us would agree that modern research would be tedious and time-consuming without the computer. We couldn't have landed on the moon, and wouldn't even be able to keep our checkbook balances straight.

But while we like to think the computer is unique to the 20th century, many of its elements are hundreds of years old. The abacus, the first digital counting machine, was designed by the Orientals and used in various forms by the Greeks and Romans. A succession of wheels, machines, engines, arithometers and calculators followed.

What were some of these devices, and how have they led us to where we are today?

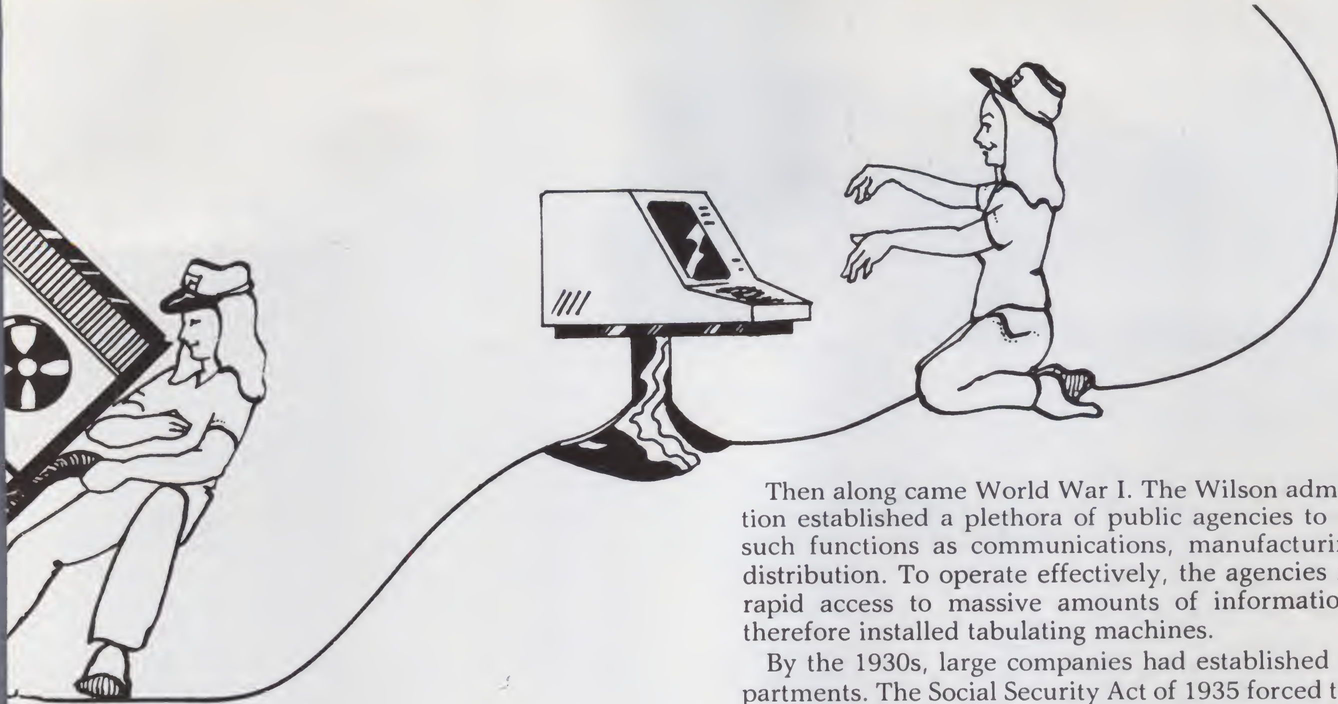
The First Calculators

The slide rule appeared in England, France and Germany in the 17th century. We know that Sir Isaac Newton used the slide rule for equations. It was to become the primary workhorse of scientists for centuries, until displaced by the pocket calculator in the 1960s.

Blaise Pascal, a leading mathematician and philosopher, developed the first real calculating machine in 1642 when only 19 years old. The machine was operated by dialing a series of wheels bearing the numbers zero through nine around their edges. He used the tally-and-carry technique still used in modern computers; look him up in an encyclopedia and appreciate what he did for civilization.

Fifty-two years later, Gottfried Wilhelm Leibniz, also a well-known mathematician and philosopher, designed a mechanical device for calculating mathematical tables. His was the first machine to multiply and divide. Much

James R. Avoli, CDP, 239 Foxcroft Road, Pittsburgh, PA 15220.



more complex than Pascal's arithmetic machine, it was designed to mechanize the calculation of trigonometric and astronomical tables.

Charles Babbage, a British mathematician, developed one of several differential engines in the 19th century. It was the first of Britain's subtle but significant contributions to computer development. The engine totaled differences to produce tables for navigation, astronomy and insurance.

Babbage then dreamed up the first computer—a machine that could handle any sort of mathematical computation automatically. His analytical engine, although never constructed, included the essential parts of a computer as we know it today: a stored program, an arithmetic section, a data entry section and an output section.

Enter Uncle Sam

Dr. Herman Hollerith, a statistician for the Census Bureau in Buffalo, designed an electric tabulating machine to help with the 1890 census. The country's rapid growth was making manual compilation of statistics difficult; figures on immigration, health, literacy and employment would have been obsolete before the census was finished. Hollerith's machine was able to compile massive amounts of data electrically.

The machine included three parts: a tabulator, a sorter with compartments electrically controlled by the tabulator's counters, and a device to punch data onto cards.

Hollerith's invention was the first statistical machine to be used on a large scale and launched the information-handling revolution. Within ten years, the railroads began to use the Hollerith machines to tabulate waybills; insurance companies used them when making actuarial statistics to correlate with mortality predictions; public utilities started to convert to mechanical accounting for their customers; and businesses used it to develop more advanced cost accounting and sales analysis methods.

Then along came World War I. The Wilson administration established a plethora of public agencies to control such functions as communications, manufacturing and distribution. To operate effectively, the agencies needed rapid access to massive amounts of information, and therefore installed tabulating machines.

By the 1930s, large companies had established tab departments. The Social Security Act of 1935 forced the government to maintain employment records on millions of people, stimulating the first of many impractical bookkeeping projects: a production line to punch, verify, sort and file hundreds of thousands of cards every day.

By then, mechanical reading, writing and arithmetic machines were available separately. Clause Shannon's 1937 master's thesis at MIT used symbolic logic to design electrical switching circuits that would add two numbers using only relays and switches. The circuitry could be kept relatively simple using base two, although any number base could be designed using his methods.

George Stibitz independently built such an adder in his kitchen, despite the facilities at Bell Labs where he worked. Then he implemented the concept of excess-3 to avoid the use of complex carry circuitry. Bell's Complex Calculator, the first relay computer, was debugged by the end of 1939. In 1940, its use by several scientists heralded the beginning of time-sharing. Also that year, in a demonstration at Dartmouth, the remote computer solved problems over telegraph circuits.

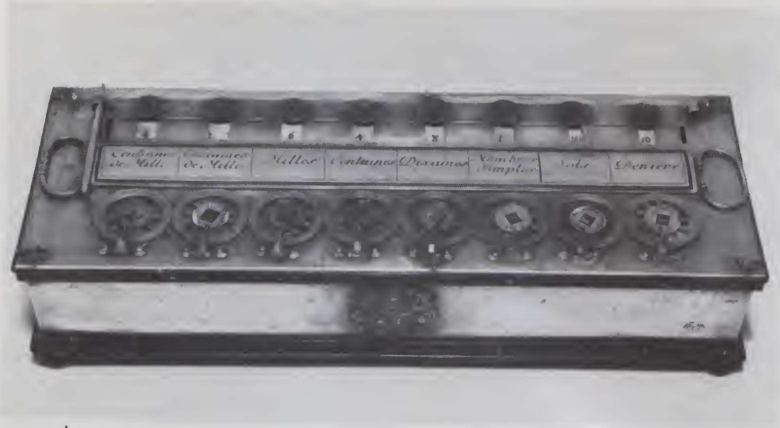
Dr. Stibitz was honored with the first IEEE Piore Citation in 1977 for these pioneering achievements, for using binary and floating-point arithmetic and for pioneering memory indexing.

During this same time, Wallace Eckert linked together different kinds of machines to perform astronomical calculations at Columbia University, and Harvard student Howard Aiken designed a new kind of calculating machine, to become known as the Mark I. It was the first automatic, general-purpose digital calculator.

Enter IBM

The International Business Machine Corp. marketed electromechanical punches in the 1930s. Beginning in 1944, IBM worked with Aiken on the Mark I. It was the largest electromechanical computer ever. They marketed the Card Programmed Calculator in 1948, which wasn't a stored program device, and was agonizingly slow by electronic computer standards.

The Electronic Numerical Integrator and Calculator (ENIAC) was the first large-scale electronic computer. The



Pascal Calculator.

pre-17th century	slide rule
1642	Pascal's arithmetic machine
1694	Leibniz's mechanical calculator
19th century	computer concept, analytical engine
1890	statistical machine in service
1935	bookkeeping assembly line
1939	relay computer (Complex Calculator)
1940	time-sharing

ENIAC was built by J. Presper Eckert and John W. Mauchly at the University of Pennsylvania. Completed in 1946, it was used by the Army for solving ballistics problems, although the machine could have had significantly wider applicability.

The project cost the Army less than $400,000, which turned out to be a real bargain. Their decision to fund the project was not cautiously planned, but because the Moore School of Electrical Engineering had a good relationship with the Army, it took the chance.

The input device on the ENIAC was an IBM card reader, and an IBM summary punch was used for output. The computer was constructed in 30 units of almost 50 panels, using 18,000 vacuum tubes, 70,000 resistors, 6,000 switches and 10,000 capacitors. When crunching numbers, it burned about 150 kilowatts of energy.

The computer was designed as a synchronous decimal arithmetic machine, with a cycle time of 200 microseconds (5000 additions per second). It could multiply two full words of ten digits (and sign) in 2.6 milliseconds. Its repertoire also included divide, subtract, compare logical and square root. Reading, writing and computing were overlapping operations.

Because the machine was hard-wired, program setup took as long as several weeks.

The need for a stored program became apparent after construction was started, and another proposal was made for a second device. The ENIAC was down about 30 percent of the time since 1946, and still managed to process over 60,000 hours of work (more than six years if the execution time were contiguous). Later it was upgraded with switch-selectable stored programs, plugboard programs and a 100-word core memory from Burroughs called the Static Magnetic Memory. A victim of superior design, ENIAC was retired on Oct. 2, 1955.

John von Neuman wrote a report in 1945 that suggested a stored program computer for the first time. The following year, the Electronic Discrete Variable Computer (EDVAC) project was started at the Moore School. Maurice Wilkes started the Electronic Delay Storage Automatic Calculator (EDSAC) in 1947 at Cambridge University in England. The EDSAC was completed before the EDVAC, therefore becoming the first operational stored-program computer.

In 1946, von Neuman started the Institute for Advanced Study (IAS) project at Princeton. Because of its random access memory and parallel binary arithmetic, this computer was faster than the serial machines.

Herman Goldstine of the IAS was awarded the 1979 Harry Goode Memorial Award by AFIPS (American Federation of Information Processing Societies) for his pioneering work in the development of the modern stored-program digital computer. He was recognized for his support and major contributions to the development of the ENIAC; his pioneering work on the logic, design and coding of electronic computers; his leadership role in the design, construction and use of the IAS machine; his significant contributions to the theory and practice of matrix computations; his encouragement of young scientists to explore electronic computers; and his pioneering of the history of computing.

After securing Army backing for the ENIAC, he became the technical director of the project. Goldstine and von Neuman, both recognizing the revolutionary potential of electronic speeds in computations, began to lay the foundation for the modern-day computer. After building the first stored-program machine at the IAS, Goldstine ran the Institute's computing center until joining IBM in 1958. There he created a group that has become a world-renowned center for the mathematics of computation.

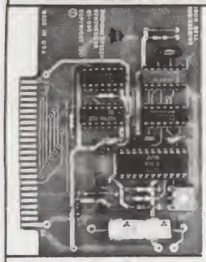
In 1947, the Whirlwind I air traffic control computer project started at MIT. The development of coincident-current magnetic core memory was its most significant contribution. The University of Manchester (England again) also began building computers in 1947. The first practical static storage system, called the Williams Tube Memory, and the first index registers were their important achievements. The Williams Tube, named after inventor F. C. Williams, was a CRT with a wire mesh over its face to help define its bit positions. Each tube stored 16 × 16 bits.

The SEAC and SWAC

Back at the ranch, the Bureau of Standards started to build two computers—the Standards Eastern Automatic Computer (SEAC) and the Standards Western Automatic Computer (SWAC). The SWAC was the first parallel stored-program computer to become operational. Engineering Research Associates proposed a drum for main memory in what was to be its ERA 1101, the ancestor of Univac's 1100 series.

Raytheon was proposing the use of a tank filled with mercury for computer memory. Harry Huskey had been responsible for the machine's design and construction and finally

SPEECH SYNTHESIZERS



APPLE II®

NEW

PARALLEL INPUT

Now you can have unlimited speech for your computer!

JBE's Speech Synthesizers use the Votrax® SC-01 Phoneme Synthesizer chip. The SC-01 phonetically synthesizes continuous speech of unlimited vocabulary. The SC-01 contains 64 different phonemes and 4 levels of inflection accessed by an 8 bit code. It requires 10 Bytes/sec. for continuous speech. Both Boards have an audio amp for direct connection to an 8Ω speaker.

Documentation includes basic user programs, a phoneme chart and a listing of coded words to help you get started.

Documentation for the Apple II® Speech Synthesizer includes a disk with many user programs.

*Because of the high cost of the SC-01 at this time, it will be sold separately. In this way, as the price goes down, we can pass the savings on directly to you.

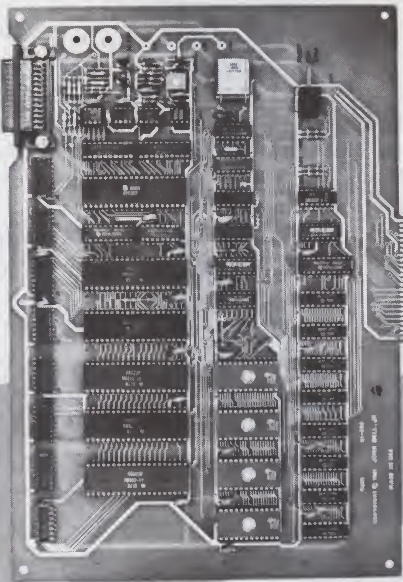
81-088	Apple II® Speech Synthesizer	$79.95
81-120	Parallel Input Speech Synth. (On board power supply, Wall transformer included)	$89.95
SC-01	Phoneme Synthesizer Chip	$59.95

EPROM EXPANSION CARD

JBE EPROM Expander for the Apple II holds six 5 volt 2716s for a total of 12K bytes of ROM. This board takes the place of the on board ROM in the Apple. It is software switchable by the same technique used by the Apple® II firmware card. Solder jumpers are for reset to the Apple ROM or 2716s on the card. (EPROMs available separately). Use JBE EPROM programmer and parralle1 I/O cards to program your EPROMS.

81-085K	Kit	$49.95
81-085A	Assm.	$59.95
81-085B	Bare Board	$39.95

JBE I MICROCOMPUTER



JBE's 7 ¾" x 11 ¾" 6502 base Microcomputer has the capacity for 16K of EPROM, 4K of RAM, 8 Parallel Ports and 1 Serial Port. Monitor and Tiny Basic are also available. The fully populated version includes:

- 1 6502 CPU
- 4 6522 VIA (8 Parallel I/O Ports)
- 1 AY5-1013 (Serial I/O Ports)
- 8 2114 RAM (4K)
- 2 2716 EPROM (monitor & tiny basic)

The partially populated version includes:

- 1 6502 CPU
- 1 6522 VIA (2 Parallel I/O Ports)
- 1 AY5-1013 (Serial I/O Port)
- 2 2114 RAM (1K)
- 1 2716 EPROM (with monitor)

Both versions include sockets for 4 2716s or 2732s, 8 16 pin sockets for I/O interfacing and a DB25 connector for RS232.

All address and data lines, power supply, RDY, interrupts, DMA, phase 1 & phase 2 clock, R/W, reset and NMI & IRQ are brought off the board to the 50 pin edge connector. (similar to the Apple II® bus)

This board also features power on reset and cassette interface.

Documentation includes 6502 programming manual & Complete doc. for the 6522 VIA. Also included is doc. for interfacing with JBE A-D & D-A Converter, Solid State Switches, EPROM Programmer & Parallel Input Speech Synthesizer.

81-030C Fully Populated JBE I	$349.95
81-030M Partially Populated JBE I	$249.95
81-030B Bare Board JBE I (Doc.Incl.)	$ 89.95
2716 EPROM (with monitor)	$ 19.95
2716 EPROM (with tiny basic)	$ 19.95

A TO D CONVERTER

NEW

JBE's 16 channel A-D Converter plugs into your Apple II® Computer. It uses an ADC0817 which Incorporates a 16 channel multiplexer and an 8 bit A-D Converter. The 16 inputs are high impedance and the voltage range is 0 to 5 volts. Conversion time is <100 μsec. The resolution is 8 bits or 256 steps, linearity is ± ½ step. Two 16 pin DIP sockets are used for input, GND & reference voltage connections. There are 3 single bit TTL inputs. Doc. includes sample program.

81-132B Bare Board	$29.95
81-132K Kit	$69.96
81-132A Assm. & Tested	$89.95

REAL TIME CLOCK

NEW

JBE's Real Time Clock Card for the Apple II® computer has automatic leap year calculation, month & date and can time down to .1 sec. It uses an MM58174 Real Time Clock chip, has interrupt capability and has separate time intervals of 60 sec, 5 sec & .5 sec. An on board 2716 holds 8 user programs which are switch selectable. A 32768 Hz crystal is used for accuracy. On board battery will keep correct time for up to one year.

81-135B Bare Board	$ 39.95
81-135K Kit	$ 99.95
81-135A Assm. & Tested	$119.95

CRT CONTROLLER

This intelligent CRT Controller uses an 8085A CPU & an 8275 Integrated CRT Controller. It features: • 25 lines (80 Char./line) • 5x7 dot matrix • Upper & lower case • two 2716s (controller & char. generator) • serial interface RS232 & TTL • baud rates of 110, 150, 300, 600, 1200, 2400, 4800 & 9600 • keyboard scanning system • unencoded keyboard is req'd • uses +5V & ±12V power supplies • Doc. includes program listing & composite video circuit.

Bare Board only (Doc. incl.)	$39.95
Programmed 2716s	$19.95

ICS

6502	$9.95
6522	$9.95
Z80 CPU	$9.95
Z80 PIO	$9.95
2716	$14.95
2716 Programmed	$19.95

JOHN BELL ENGINEERING, INC.

ALL PRODUCTS ARE AVAILABLE FROM **JOHN BELL ENGINEERING** • P.O. BOX 338
REDWOOD CITY, CA 94064 • ADD SALES TAX IN CALIFORNIA • ADD 5% SHIPPING & HANDLING
(415) 367-1137 10% OUTSIDE U.S.A.

MC VISA

1940	teleprocessing
early 1940s	automatic general-purpose calculator
1945	idea for stored-program computer
1946	large-scale electronic computer
1946	random access memory
1947	stored-program computer
1947	coincident-current core memory

made the decision to use the Williams Tube Memory. Thirty-six tubes provided a total memory capacity of 256 36-bit words (plus sign). By today's megabyte standards, it was insignificant, but in August of 1950 it made the SWAC one of the fastest computers in existence. The 1978 NCC Pioneer Day Award went to Huskey; in keeping with the event, the award was one of the now-dismantled SWAC's Williams tubes.

The SEAC was the first stored-program computer to run in the United States. It went into operation in 1950, using a mercury delay line memory. Other types of memory were added later.

The Eckert-Mauchly Computer Corp. delivered the Univac (Universal Automatic Computer) to the Bureau of the Census in June 1951 for the 1950 census. The Univac I became the first commercially available computer, setting the stage for the Eckert-Mauchly Division of the Remington Rand Corp. In 1952, they acquired Engineering Research Associates, which had already developed the ERA 1103 (a versatile scientific computer) and the best rotating drum memories.

With today's advanced pocket machines, the distinction between computers and calculators becomes fuzzy. The electronic slide rule is now being replaced by programmable pocket computers, more versatile than the first commercial computer. Companies that manufacture both devices, such as Texas Instruments and Hewlett-Packard, preserve the distinction in advertising literature.

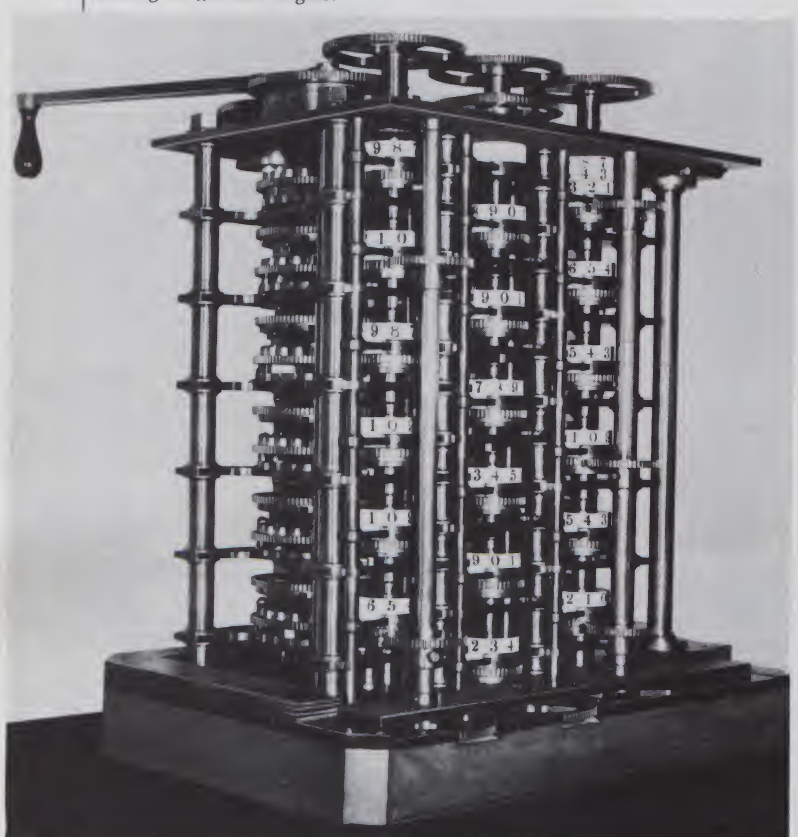
It is difficult to cite the specific attributes that relegate a device to one class or the other. Before 1953, however, the problem went the other way: everything could be called a computer. Hence, the computer and the calculator were (at the time) synonymous.

During the Korean War, IBM announced a large-scale scientific computer called the Defense Calculator. The IBM 701 was delivered in 1953; the IBM 702 was delivered in 1955 for the commercial users, with 10,000 characters of electrostatic Williams Tube Memory.

By 1955, many agreed that the Univac computers were superior to those of IBM. IBM delivered the first 705 in 1955, less than a year after the 702. The first Univac II wasn't delivered until 1957, and these two critical years were enough for IBM to forge into the lead that they enjoy today.

1960 was the time when many organizations were moving from IBM 704s to 7090s (some through the 709 phase), while others considered the Univac 1105 and the Control Data 1604. A number of unique systems were each attempting to move the marketplace in a special direction. It was an active period; computers announced included RCA's BIZMAC, Honeywell's Datamatic 1000, Raytheon's RAYDEC, TRW's RW400, Univac's LARK, IBM's STRETCH, Westinghouse's ILLIAC IV, Burroughs' B5000 and Bendix's G-20.

Babbage Difference Engine.



The Generation Gap

The first generation of digital computers was slow, with small internal memories, able to process one job at a time and dependent on magnetic tape for the storage of bulk data. They were characterized by vacuum tubes, which represented such problems as reliability, expensive maintenance, heat dissipation and vast requirements for power. These are the machines that gave the world its impressions of the computer as the electronic brain.

The second generation evolved between 1958 and 1965. The transistor was its major characteristic. During this generation, channels, multiprogramming and time-sharing started to affect the data-processing community. These machines provided larger internal memories, faster execution times, higher reliability, less heat, lower power consumption and generally more bang for the buck.

The basic architecture was the same as the previous generation, because magnetic tape was still the primary medium for data files. Data communications was introduced for machine-to-machine interaction, but the computer still usually processed only one job at a time.

The third generation followed in 1965 and was characterized by the microminiature integrated circuit. It significantly reduced the size, improved the reliability, reduced vast power consumption and reduced production costs.

These machines had bigger internal memories, allowing much larger, more complex programs to be written.

An interrupt capability allowed the computer to be shared among jobs for short intervals, thus encouraging more extensive use of data communications. Mass storage devices acquired much more capacity, providing data online to the computer at costs that approached those of magnetic tape. However, except for interrupt handling, the architecture had not been improved significantly.

Then, in 1973, Federal Judge E. H. Larson of Minneapolis ruled that the patent held by the Sperry Rand Corporation was not valid. Honeywell had charged that Sperry had fraudulently procured the ENIAC patent and had restrained and monopolized the EDP industry. They maintained that because the patent was not granted until 1964, a delay preventing the invention from promoting the progress of science and the useful arts had been created. They claimed the design was in public use before the patent was granted, and they maintained that others besides Eckert and Mauchly should be regarded as the inventors of the electronic digital computer. Mauchly is said to have derived his ideas from John V. Atanasoff, a physicist who had constructed a prototype to solve differential equations. He had visited Atanasoff and had seen the device that had been developed but had never worked properly.

Mauchly and Eckert were able to fund and

Harvard Mark I.

build such a device, so despite the court's decision, they are entitled to be considered the inventors of the first fully operational electronic digital computer.

Programming Languages

Many problem-oriented languages (POLs) have been developed over the years.

In the early 1950s the University of Illinois hosted the programmers of the ILLIAC, including computer pioneers Wheeler and Gill of Cambridge, England. Most of them were coding their numeric machine commands and addresses agonizingly to fit their programs within fixed address locations in the 1024-word memory. The Englishmen began to generate massive amounts of code, written in symbolic

1947	index registers
1947	static storage
1948	Card Programmed Calculator
1950	parallel stored-program computer
1950	first U.S. stored-program computer
1951	first commercial computer

Time frame	Milestone	Credit goes to
pre-17th century	slide rule	Orientals
1642	arithmetic machine (+ and −)	Pascal
1694	mechanical calculator (* and /)	Leibniz
19th century	computer concept, analytical engine	Babbage
1890	statistical machine in service	Hollerith
1935	bookkeeping assembly line	Social Security Admin.
1939	relay computer (Complex Calculator)	Stibitz
1940	time-sharing	Bell Labs
1940	teleprocessing	Bell Labs
early 1940s	automatic general-purpose calculator	Aiken
1945	idea for stored-program computer	von Neuman
1946	large-scale electronic computer	Eckert & Mauchly
1946	random access memory	Goldstine
1947	stored-program computer	Wilkes
1947	coincident-current core memory	MIT
1947	index registers	U. of Manchester
1947	static storage	Williams
1948	Card Programmed Calculator	IBM and Aiken
1950	parallel stored-program computer	Bureau of Standards
1950	first U.S. stored-program computer	Bureau of Standards
1951	first commercial computer	Eckert-Mauchly Computer Corp.
early 1950s	subroutine and symbolic assembler	Wheeler and Gill
1958-1965	second generation machines	the transistor
1965-1976	third generation machines	the integrated circuit
1976-1979	fourth generation machines	large-scale integration and microcomputers
1979-early 1980s	fifth generation machines	cryogenic electronics, fiber optics communications, very large-scale integration

early 1950s	subroutine and symbolic assembler
1958-1965	second generation machines
1965-1976	third generation machines
1976-1979	fourth generation machines
early 1980s	fifth generation machines

form, using what was called a one-pass assembler. The Yanks stuck to their machine coding, but slowly agreed to use the innovation.

The message was clear—programming productivity could be enhanced by the use of the symbolic assembler and the subroutine, which was also the invention of these English scholars. Formalized with the assistance of Wilkes, their fundamental book, *A Comprehensive Set of Mathematical Subroutines*, was published in 1951.

The diversity of machines and implementation languages in the '50s and '60s, together with the need to retain flexibility in implementing systems, led to the search for either the one machine-independent and universally accepted language or for methods and techniques to make all languages readily available to all machines. The first of these alternatives resulted in many efforts to generate variations of FORTRAN and COBOL, such as ALGOL and PL/1.

But have the compilers flowed like wine since then! They include ALGOL, APL, BASIC, CL/1, CLIP, COMIT, COMMERCIAL, EASYTRIEVE, FACT, FORTRAN, JOVIAL, LISP, MAD, MARK IV, NELIAC, Pascal, PL/1, RPG, RSTS, SURGE, SNOBOL, TABLEMAKER, UMAP, WATFIVE, WATFOR, XTRAN—and their variations.

This leads to the question concerning the search for the universal computer-oriented language (UNCOL). In the '60s, organizations were devoted to the unrewarding task of structuring a language that would adequately express the capability of any one of a number of machines. Each different machine would have a compiler written for UNCOL. The POL could be implemented in UNCOL once and, via the many UNCOL compilers, would be available for all UNCOL-served host computers.

The hardest blow to the UNCOL concept came as a result in compiler-writing breakthroughs. Prior to the '60s, compiler financing approached a half-million dollars and took up to three years. In the early '60s independents were generating compilers for one-tenth the cost and one-fifth the time.

The Air Force wouldn't let UNCOL become a dead issue, however. UNCOL objectives for JOVIAL were targeted onto IBM's 7090 and 360 machines, as well as several other manufacturers' mainframes.

So now the technology of the fourth and fifth generations have emerged in the form of computer products for consumers. Thanks to the American space program, which essentially funded the development of microelectronics, the move into these generations can be attributed to the incredible cost reductions, the microprocessors, the gargantuan memories and, at last, the home computer. ■

Take a bite...

...of Apple Orchard!

Your Apple will love you for subscribing to APPLE ORCHARD! An official publication of the International Apple Core, APPLE ORCHARD has all the information you need for your Apple. In addition to outstanding feature articles, we present new product information, product and book reviews, program listings, articles and news from member clubs (maybe yours!), and IAC information, including club listings, sponsor listings, and upcoming events.

Please enter a subscription to APPLE ORCHARD for:

NAME ______________________

ADDRESS ______________________

CITY __________ STATE __________ ZIP __________

Subscription rate: $10.00 for four issues
$5.00 additional required for first class postage to Canada, Mexico, APO, and FPO addresses
$10.00 additional required for air mail postage to overseas and other foreign addresses

☐ Check or money order enclosed ☐ VISA ☐ MasterCard

No. ______________________ Exp. __________

Reply to: **Apple Orchard Subscriptions** ✓387
P.O. Box 1493
Beaverton, OR 97075

Subscribe today! You . . . and your Apple . . . won't want to miss a single issue!

✓ *See List of Advertisers on page 210*

For those times when you need a good look at a signal, Heathkit has the answer—the Model IO-4205 5 MHz Dual Trace Oscilloscope.

Scope It!

By Tom Lukers

The Heathkit Model IO-4205 is a dual-trace, general-purpose utility scope, designed for hobbyists, technicians and engineers. It costs about $400 and boasts features usually found only in more expensive scopes.

The Model IO-4205 has a 5 MHz vertical bandwidth, dual-trace triggered sweep and 1 megohm input impedance. Vertical sensitivity is 10 mV per centimeter. Vertical gain is switch-selectable in 11 calibrated ranges, from 10 mV per division to 20 volts/division. And the time base is switch-selectable over seven ranges, from .2 μs to .2 seconds. Within each range, the time base is vernier-controlled with a smoothly acting pot.

The 5 MHz bandwidth makes it adequate for work on most small computer systems and even color television sets. Although the 1 megohm input impedance might possibly load down some critical circuits, this is a rare possibility with home computer systems. The built-in triggered sweep circuit means that you don't have to be concerned about synchronizing the scope with the signal under test. The scope automatically does it for you.

Although you don't always need a scope to troubleshoot small computers, they sometimes help. Frequently, a logic probe is all you need. But at other times, an oscilloscope is your only recourse.

Digital signals are binary. That is, a signal is either there or it isn't. And most of the time, that's all anyone cares about. So you can do most digital troubleshooting with a simple logic probe, or, in some cases, with just an LED and a current limiting resistor. But even in the digital world, you sometimes need to know what the signal *looks* like.

You sometimes need to know the signal's shape, rise time, fall time, width (in microseconds or milliseconds) and height, in volts or other units. Even digital signals can get so distorted that TTL devices will no longer recognize them. And there are frequently analog signals in the system that you need to watch. So what do you do when you need a good look at a signal? Simple. You use an oscilloscope. But it doesn't have to be an *expensive* scope.

Construction

Although this is meant as a product review and not a construction article, a few construction points are worth considering. First, the fact that it's a Heathkit says volumes. The Heathkit people have decades of experience in designing kits.

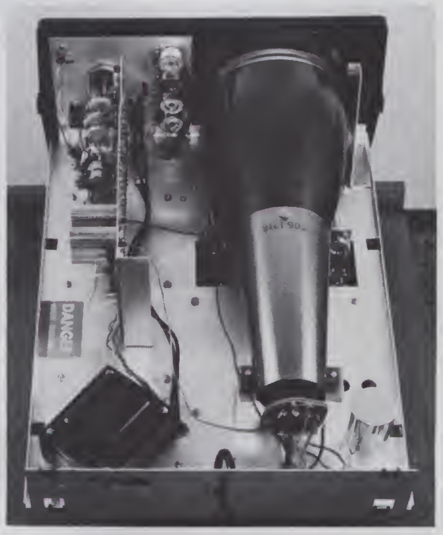
If you already know about Heathkits, move on to the next part of this article. But if you have never built a Heathkit, read on.

My first Heathkit was a little handheld multimeter, back in 1952. After more than 27 years of kit-building and working with students, technicians and engineers, I have developed a sort of "kit builder's laundry list." I've checked it informally with a number of other instructors and against what I've found in the literature and it seems to be a pretty good list. At least, the points made here are good, if not all-inclusive. So here it is.

For one thing, there are definite advantages to modular design with clean, uncluttered layout, simple wiring, and plenty of room on circuit boards and the chassis. Good (rigid) mechanical construction, quality printed circuit boards and sockets for all integrated circuits are other preferences.

In addition, when the job is done, it should look sharp, work as advertised and give rock-steady, reliable operation. And all of the screws, nuts, washers and other hardware should be there. You don't want to end up short by one or two pieces.

Good documentation is a must. And the manuals should be written to



This top view of final assembly shows the roomy, uncluttered layout. The power transformer is at left rear, with the filter capacitors to the right. The two controls to either side of the CRT center are the vertical gain controls. Mounted on the front panel just to the left of the CRT are the intensity, focus, trigger level and horizontal position controls. The horizontal circuit board is to the left.

Address correspondence to Dr. Tom Lukers, PO Box 1949, 3625 Hendrick Drive, Plano, TX 75074 (Micronet 70130,371).

tell you how to put the product together, how it works and how to use it. One of the first warning signals for any kit is when it becomes apparent that the manual was written to impress you. It usually doesn't. One that was written to impress usually doesn't tell what you need to know about the product.

Happily, the Heathkit Model IO-4205 rates very high on all of these points. On a scale of 10, I would rate it an 8 or a 9 in all categories except the documentation. They rate a 10 on that.

Some reviews have referred to Heath documentation as "overkill." But I don't believe that it is possible to give too much good, accurate information with a product that is designed to be built by anyone from a novice to a professional engineer.

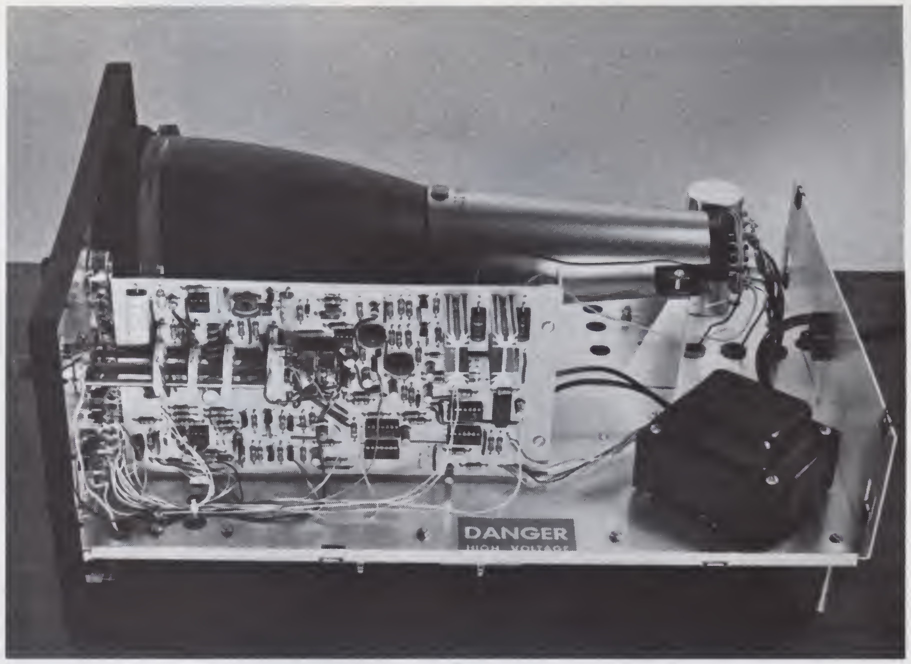
If you're a newcomer to electronics and have never built a kit, I would recommend that you build a couple of smaller kits before tackling the Model IO-4205. Not that it's all that difficult. But if you have no experience in soldering, it is a good idea to get some before tackling such a project. From working with hundreds of students on building projects, I believe that soldering technique can contribute more to the success (or failure) of a project than any other single thing.

But if you have successfully built a couple of smaller kits, then I would recommend the Model IO-4205 without any reservations.

I built mine in one week of evenings plus one entire weekend. I work slowly and carefully. And I've found that it pays to follow the Heath directions to the letter. If you work carefully, follow instructions and pay attention to details, you will probably end up with a great scope that will give you years of trouble-free service. Not only that, but following details carefully makes it much easier to backtrack if things don't work right at first.

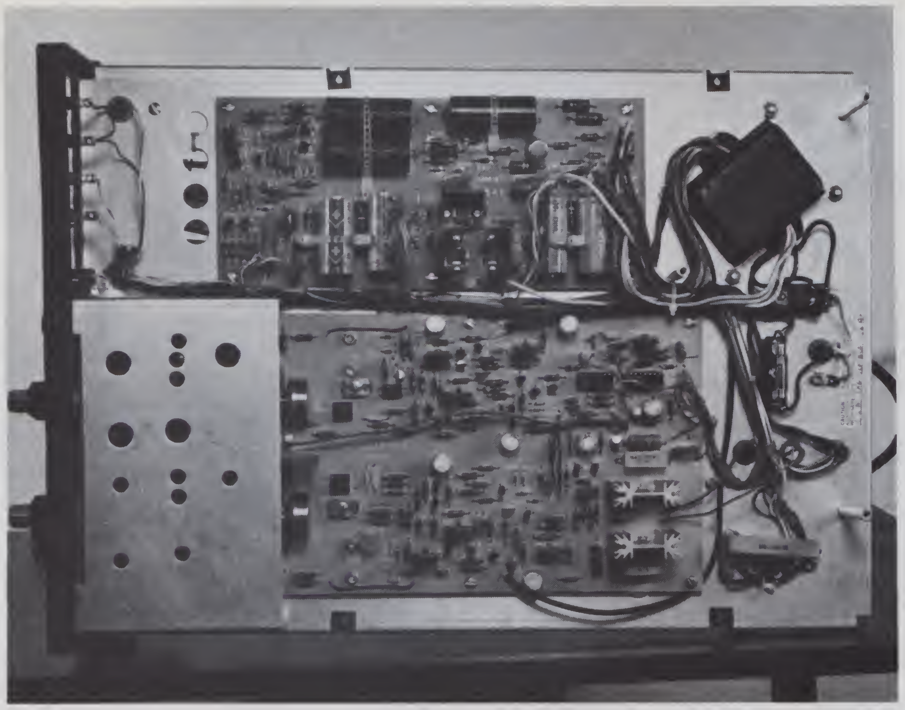
The kit consists of three printed circuit boards. They are the vertical amplifier, horizontal amplifier and power supply.

The boards are clearly printed with identifying marks for all components. Each board is neat and uncrowded. Mechanical mounting is strong and rigid. The prefabricated wiring harness greatly simplifies construction. The chassis is roomy enough to make it convenient during servicing but small enough to make a

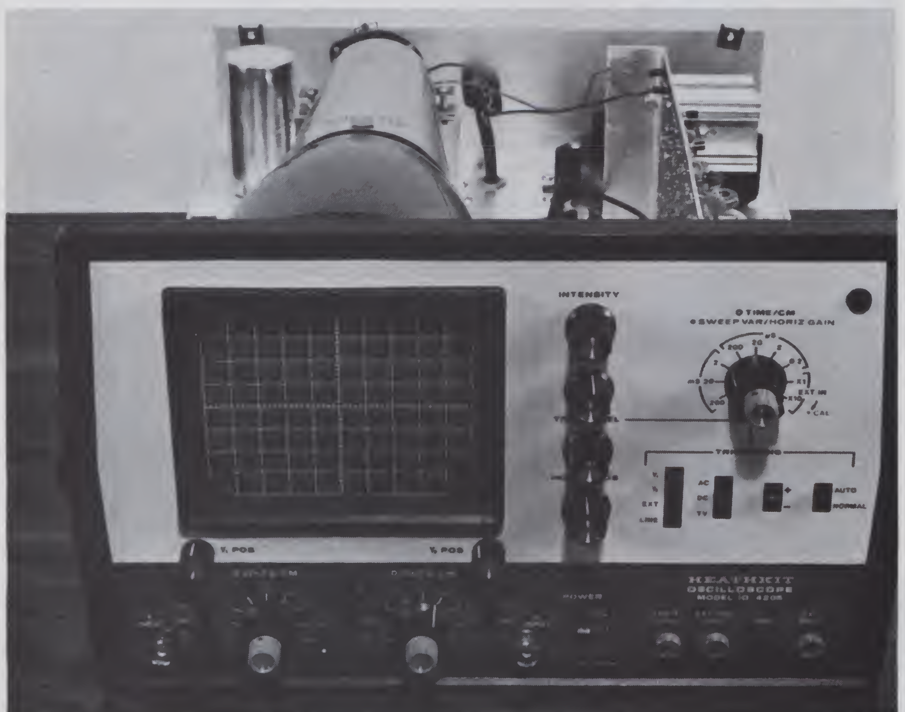


This view shows placement of the horizontal circuit board, the CRT and the power transformer. The horizontal circuit board contains the time base generator, the horizontal amplifier and the horizontal deflection amplifier for the CRT.

My daughter, Beulah, is a junior at Brigham Young University, majoring in Russian and Chinese, with a math minor. She is an honors student and has recently become interested in applications of microcomputers to language translation. Here she is using the Heathkit Model IO-4205 scope and the Heathkit digital logic trainer in some exploratory experiments in digital logic.



This is the bottom view of the scope. The power supply circuit board is at the top with the bottom of the power transformer at right. The vertical amplifier board is at the bottom. It contains the vertical amplifier and the vertical deflection amplifier circuits. The vertical amplifier circuits are shielded to prevent interaction from stray magnetic fields. The holes in the shielding are for making screwdriver adjustments during calibration. Although the wiring harness looks complicated, it really isn't that bad. The harness is prefabricated to simplify construction.



This is the Model IO-4205 during final assembly. The time base/horizontal gain controls are at the upper right; directly underneath are the four triggering slide switches mentioned in the text. The vertical column of four switches just to the left of the time base switch is for intensity, focus, trigger level and horizontal position. Just below the CRT are the vertical position controls for each channel, with the vertical gain controls and BNC input connectors just below. Each channel input can be direct coupled, grounded or ac coupled, using selector switches. Just behind the front panel can be seen the horizontal circuit board on the right, with the CRT in the center.

compact, convenient instrument.

The construction and operation manual includes clear, detailed instructions for building, calibrating and operating the scope, plus a theory of operation and detailed description for each circuit. In addition, there are six pages of semiconductor identification charts and parts lists.

Heath provides a large schematic diagram and a 38-page illustrated booklet in addition to the excellent construction/operation manual.

The schematic is well laid out and clear enough to read even if your eyes are not as good as they used to be. The detailed sketches in the illustration booklet make it easy to identify the components, to see where to place them and how to orient them. One neat technique used by Heath is to show each construction area with all the parts stripped away except those being installed. Thus, the same part of the board or chassis may appear several times, but each time it only shows those parts being installed. This makes it easy to see exactly how one part is oriented with respect to the others.

The construction and operation manual includes clear, detailed instructions.

There is just one thing that I don't like about the IO-4205, and that is the way they located the four slide switches for triggering. These switches are AUTO/NORMAL, (+)/(−), AC/DC/TV and Y1/Y2/EXT/LINE. They are mounted under the plastic false cover on the front panel. That is, they are sandwiched between the metal front panel and the plastic silk-screen panel showing the control names. The screws holding these switches are located under the silk-screen panel. They tend to loosen up after a while, and it is a tedious job to tighten them again. This is a minor irritation, however. It has been a nuisance only about three times in as many years.

Heath provides two coax cables for the scope, each fitted with alligator clips for the signal and ground lines.

High-frequency probes are available from Heath for $29.95 each. I have both kinds. But the cables provided with the kit have been adequate for most of my uses.

Applications

The Model IO-4205 is useful for testing a wide range of equipment. And if you know nothing about using a scope, Heath's Self Study Course will help get you up to speed.

I built this scope in 1977. And I've used it to check out, calibrate and troubleshoot several computers, an H19 terminal, an H14 printer and several 72 MHz transmitters and receivers for radio-controlled model airplanes.

This scope makes troubleshooting the R/C receivers and transmitters a snap. Although the 72 MHz operating frequency is out of the scope's design specifications, the scope is still quite useful. In the receiver, for example,

This scope makes troubleshooting the R/C receivers and transmitters a snap.

the intermediate amplifier frequency is 456 kHz. This is well within the scope's range of 5 MHz. So the only times you can't clearly see the signals are in the rf amplifier and converter stages. But even in these stages, it is usually enough to know whether the signal is there, without knowing what it looks like.

Here's how you can use this scope on your computers: With the system working properly (*before* trouble starts), you can look at the waveforms for key signals, noting the shape and amplitude. Then when you have trouble in the system, if you don't get those same signal shapes and sizes, it's a good bet that you're on the right track. This way you can isolate the trouble to the faulty part of the system. Similar techniques will work with almost any equipment.

So there you have it. The next time you have trouble in your home computer and don't know where to start, don't *just stand there*. Scope it, with the Heathkit Model IO-4205 scope! ■

✓See List of Advertisers on page 210

Combining minicomputer features in a microcomputer package, the Hewlett-Packard HP-85 is sure to be a big hit for business/science applications.

What's So Super About the HP-85?

By Russell King

I was first introduced to the Hewlett-Packard HP-85 at Microshack, the local computer store. I was not initially impressed; it looked like just another microcomputer with a pretty face. But I quickly changed my mind after Dave Claypool, Microshack's manager, showed me some of its capabilities. This machine's beauty was more than skin deep.

The Hewlett-Packard HP-85 is really a minicomputer in a microcomputer package. It contains nearly every feature that microcomputer critics have wanted in a machine. It is an all-in-one package containing a 92-key keyboard, five-inch (diagonal) display, thermal printer, DC100 tape cartridge drive and BASIC in ROM. The machine can be expanded to include extra memory, more BASIC in ROM, disk drives, printers, graphic tablets, plotters, interfaces and literally a hundred devices. Hewlett-Packard rounds out the accessories with a number of software packages designed for scientific/engineering applications, text-editing and games.

This machine will especially attract the attention of those people with scientific, engineering or instrument controller applications. The features in the basic package are superb, and include graphics, a built-in tape unit that is as fast and as versatile as a disk unit, extra function keys, a great BASIC and good documentation. In short, the HP-85 has nearly everything one could want in a small microcomputer.

Address correspondence to Russell King, 15 Fyfe St., Regina, Saskatchewan S4X 1J7, Canada.

Owner's Manual

The owner's manual is the best that I have seen. It is well-written and covers every aspect of the machine. In addition to a good table of contents, the manual has a comprehensive index and appendices. It also has good, relevant sample problems with solutions to help the owner learn the HP-85's functions.

Keyboard and Display

The keyboard has a professional feel, and with its 92 keys offers more capabilities than any other microcomputer on the market. Another welcome feature, but one that requires getting used to, is the line mode operation of the keyboard—the computer will accept the entire line that the cursor is placed on when the enter key is pressed. Thus, if you wish to modify a line, you merely type in the changes and re-enter the line, instead of retyping the entire line. This feature makes editing easier.

The key functions can be grouped into four function areas: the alphanumeric keyboard, the numeric keypad, function keys and system control keys.

The alphanumeric keys are laid out in a typewriter format, except that uppercase characters are normally generated, and the shift key (or the caps lock key) generates lowercase characters. This can be reversed (lowercase generated normally, with uppercase generated with the shift key) by using the FLIP command.

The Hewlett-Packard HP-85—"The machine's beauty is more than skin-deep."

The HP-85 calls the return key END LINE, and the break key PAUSE.

The numeric keypad provides a quick means of entering numeric data, and can function as a direct-entry calculator without the need to type PRINT statements. Unfortunately, the enter key (END LINE or RSLT) is poorly placed, lessening its advantage.

The function keys provide a variety of useful features. Four keys control cursor positioning, which in conjunction with the edit and line mode features make program modifications a snap. Other keys will erase a line from the cursor position to the end of the line, automatically number the lines of the program as you enter them, delete characters, allow insertion of characters, clear the screen, backspace the cursor or scratch (erase) your program.

Also included on the keyboard are special function keys that can be defined and tested by your program. A program can assign a descriptive label to each special function key which will appear at the bottom of the display.

Finally, there are several BASIC keyword keys: RUN, LIST, CONTINUE, LOAD, DELETE (to delete program lines) and a few others. There is even a key that lets you single-step through your program.

Rounding out the keyboard features, the system keys rewind the tape cartridge, copy the display to the thermal printer, scroll the display up or down, advance the paper in the printer, or initiate a system self-test.

Although the CRT screen is small, it is quite readable. It has a graphics resolution of 256 by 192 and an alphanumeric resolution of 16 lines, with 32 characters per line. An interesting feature of the display is its 64 line buffer. By scrolling up or down you can display any 16 contiguous lines in the buffer. The graphics buffer is separate from the alphanumeric buffer.

Tape Cartridge Unit

The tape cartridge system is one of the best features of the HP-85; it is the fastest and most versatile tape system of any microcomputer system. One tape cartridge can hold up to 195K of program information or 210K of data. Programs and data can be mixed on a tape cartridge.

Each tape cartridge has its own catalog which keeps track of up to 42 files. The catalog can list the file names, the file types (program, data or binary), the number of bytes per record, the number of records in the file and the file sequence number assigned by the computer. You can use the built-in file security features of the HP-85 to prevent listing or editing, duplicating or overwriting, or even to prevent the program from appearing in the catalog. The file security feature is very easy to use.

The best feature of the tape cartridge is its file-handling capabilities, which are equal to those found on disk systems. It can handle sequential-access, direct-access and random-access files.

The DC100A tape cartridge is not the standard tape cassette; it is heavier and faster. Although this cartridge is much more expensive than cassettes and more difficult to obtain, its advantages in speed and reliability make it worthwhile. The HP-85 cartridge can be read at ten inches per second and one can be rewound in 29 seconds. When a file search is in progress the tape advances at 60 inches per second. You don't wait long for a program to load!

Other Features

The printer is a quiet thermal printer with the same resolution as the display, and can handle alphanumerics, graphics and even strip charts. It can print two lines (64 characters) per second, and will copy the display in eight seconds. Anything that can be displayed on the screen can be copied to the printer.

The built-in beeper is easy to use (use the BASIC command BEEP). Both tone (frequency or pitch) and duration are controllable, and the beeper can be used to generate computer music.

The HP-85 also has a built-in real-time clock and three programmable timers. The clock is similar to the large computer clocks, in that it keeps track of the time in seconds since midnight (IBM clocks keep track of time in hundredths of seconds since midnight). The programmable timers can be independently controlled to provide interrupts at intervals ranging from half a millisecond to just over 27-3/4 hours. As with the other features, these timers are easy to use.

Accessories and Expansion

Many options are available for the

✓See List of Advertisers on page 210

HP-85. RAM memory can be doubled with the 16K memory module, allowing a total RAM memory of 32K. ROMs can be added to enhance the built-in BASIC and control peripherals to simplify expansion. Currently, ROMs for matrix math are available, as are ROM controllers for a plotter, a 132-column printer and disk drives.

Hewlett-Packard also has four interface modules available: an RS-232 interface, a GP-IO (input/output) interface, a BCD (binary coded decimal) interface and the HP-IB interface. The HP-IB interface is the IEEE-488 interface standard, which Hewlett-Packard uses on almost all its minicomputers. Hewlett-Packard has 199 products that can be attached to this bus.

HP also sells blank tape cartridges, rolls of paper for the printer, a carrying case for the HP-85, cartridge/manual holders and ROM drawers (devices that hold the ROMs; one ROM drawer will hold six ROMs).

Software Packages

Hewlett-Packard, recognizing that a computer is not useful without software, has several software packages on tape cartridges, which they call Application Pacs. The Pacs currently available are primarily designed for scientific and engineering applications, and include General Statistics, Mathematics, Circuit Analysis, Linear Programming, Waveform Analysis, Basic Statistics and Data Manipulations, Regression Analysis, BASIC Training self-study course, Finance, Text Editing and Standard Pac (a number of programs that use the HP's features). Hewlett-Packard is busy expanding the number and variety of Application Pacs.

The Application Pacs come with a manual describing the programs on the cartridge, but these manuals are not as good as the owner's manual. I looked over the Math Pac, and found programs dealing with solving simultaneous equations, triangle solutions, integration, differential equations, Chebyshev polynomials, fast Fourier transforms, computation of hyperbolics and complex number operations.

Reliability

If the HP-85's reliability matches other Hewlett-Packard products, it will provide years of trouble-free service. The company I work for owns a desk-sized HP-250, which has been loaded into a truck and driven around the province winter and summer (temperatures vary from about −40 to 100 F or more), has been dropped once and has never required more than casual maintenance.

BASIC

The HP-85 BASIC has more features than the BASIC you get with most microcomputers; in fact, it compares favorably to the enhanced BASICs found on mainframe computers.

The built-in functions available on this machine emphasize that it was designed for scientific/engineering work. A larger number of trigonometric functions are featured with this BASIC than with most other BASICs. They include CSC (cosecant), SEC (secant), SIN (sine), COS (cosine), TAN (tangent), ASN (arcsine), ACS (arccosine), COT (cotangent), ATN y/x (arctangent of y/x for conversion between polar and rectangular coordinates), DTR (degrees to radians), RTD

(radians to degrees), DEG (for operations in degrees), GRAD (for operations in grads) and RAD (for operations in radians). Other functions, including multiple-line functions, can be defined using the DFN (define function) command.

You can do regular or modulo arithmetic (clocks use modulo arithmetic; i.e., four hours after 10 PM is 2, not 14). You can also check signs, remainders, integer portions of a number and fractional portions of a number (i.e., the integer portion of 3.5 is 3, while the fractional portion is 5) with single commands.

The random number generator uses a seed from which random numbers are generated. The seed lets you get a more random selection of random numbers. (The OSI C1P micro generates the same random numbers each time it is first turned on—your micro probably does, too.)

Other features include three built-in constants: PI, EPS (the smallest positive number possible on the HP-85) and INF (the largest positive number possible). The smallest positive number is indeed small, being approximately $1/(10^{500})$, while the largest number is approximately 10^{500}. Any real number can have up to 12 significant digits, while integers can have up to five significant digits.

The HP-85 also has four number comparison commands that can be useful: FLOOR(x), CEIL(x), MAX(x,y) and MIN(x,y). FLOOR will return the largest whole number less than x (e.g., CEIL 3.5 is 4). MAX and MIN will return the larger or smaller of the two numbers x, y, respectively.

Three commands let you use program overlay (see "Overlay Programming" by Robert A. Peck, *Microcomputing*, October 1980, p. 208): COM, CHAIN and UNCHAIN. COM specifies which variables will be common to both the original (calling) program and the program being called. All variable values in the COM area will be saved; those variables outside the COM area will not. CHAIN is the command that calls the new program, and it can appear anywhere in a program any number of times. The UNCHAIN command restores the original program to memory, but any variable values outside the COM area will be lost.

Arrays and Strings

HP-85 BASIC handles strings and arrays differently than most microcomputer BASICs (like Microsoft) do. Character strings that are more than 18 characters in length (up to the theoretical maximum length of 32K) must have their lengths defined via a DIM statement. Thus a 2000 character string would be defined with DIM A$(2000). The crucial difference is this: character arrays cannot be defined. For example, DIM A$(5) defines a character string five characters in length, not a character array with five elements. I suspect that Hewlett-Packard will be announcing a ROM for handling character arrays in the near future.

Hewlett-Packard has compensated for the lack of character arrays by providing several powerful string handling abilities. Some of these commands convert numbers to characters and vice versa which may allow you to convert a character to a number, which can then be stored in a numeric array.

✓See List of Advertisers on page 210

String manipulation is also easy, since any character in a string can be accessed by specifying the first and last positions you want in the string. Thus PRINT A$(5,6) will print the fifth and sixth characters in the string A$. You can then simulate character array handling with proper programming. Unfortunately, the effort involved in converting existing programs to run on the HP-85 can be a major effort.

Edit and Debug Features

Hewlett-Packard's editing features are pretty useful. You can renumber the lines in a program; delete blocks of lines; insert, delete or replace characters on a line (remember that the line does not have to be retyped in its entirety), and even copy lines (change the line number of the line to be copied to the desired line number and press the end line key).

Another impressive feature of the HP-85 is its debug ability. You can trace the execution of your program statement-by-statement, trace the changes in one or more variables as they occur, or trace everything—the order of the execution of the statements, when each variable changes and what it changes to. You can, in addition, define error exit routines that tell you which statement caused the error, what type of error occurred and what to do about it. You can instruct your program to ignore errors and continue or to stop. These features can reduce your debugging time.

Graphics

The graphics software is excellent. It matches expensive commercial plotting software in most respects, and even provides some extra functions. The graphics display is capable of a resolution of 256 by 192 dots, with a black, white or mixed background color. You can plot points, or define and plot special characters and symbols.

The HP-85 has 16 plot commands that make graphics simple to use. You can generate x or y axes with one command and label your graph with the LABEL command (the labels can be parallel to the x or y axis, controlled with the LDIR or label direction command). An x,y coordinate system is used, and the coordinates can be absolute—point to a fixed position on the screen—or can be drawn in relation to the origin (intersection of the x and y axes).

The BPLOT command lets you define an 8 by 12 dot symbol, which can then be placed anywhere on the graph. The seven commands, PEN, PENUP, PLOT (x,y), MOVE (x,y), IMOVE (x,y), DRAW (x,y) and IDRAW (x,y), in conjunction with the LABEL command, provide the control for drawing most graphs. Commercial plot packages use PEN and PENUP commands to put the pen on the graph paper and lift the pen. This same concept is used for the HP-85 graphics. PLOT (x,y) will plot a point at the fixed screen location defined by (x,y); MOVE (x,y) moves the "pen" to the location defined by (x,y); while DRAW (x,y) will draw a line from the current position to the position defined by (x,y). IMOVE and IDRAW operate similarly, but the (x,y) position is referenced to the origin (if one exists).

You can also SCALE the x and y axes separately; the SCALE com-

mand compresses or expands the x and y axes. Thus you could have x axis intervals of 1, 2, 3, 4, etc., and y axis intervals of 100, 200, 300, etc.

The HP-85 graphics display is separate from the alphanumerics display, letting you switch back and forth between your program listing and the graph. The graph can be copied to the printer using the copy key. You can also input data while using the graphics. The graphics capabilities are further enhanced with the addition of a plotter and the printer/plotter ROM.

Drawbacks

The HP-85 is a great little machine, but it does have some disappointments.

First, the price. At $3250 (US), the HP-85 is expensive. A potential buyer must carefully weigh the benefits of the machine against its price.

Second, it has no character arrays. Although this problem can be circumvented, it will be a factor considered by potential buyers.

Third, it has a good keyboard, but more thought could have been put into the positioning of some of the keys.

Fourth, although the HP-85 has the best standard editing features of any microcomputer, it still falls short of perfection. More extensive editing features would have been possible without greatly affecting the price.

Finally, the small display screen makes this a one-user computer. A larger screen would be a great improvement. The 32-column display is a big disappointment—a machine with such capabilities should have an 80-column display (the very expensive HP-9845 has these features and color graphics as well).

The HP-85 will be a disappointment to those hobbyists that enjoy using PEEKs and POKEs and USR commands—they don't exist. Hewlett-Packard has tried to construct a BASIC complex enough to make such commands unnecessary, and have succeeded to a large degree. Any shortcomings in their BASIC will probably be remedied with BASIC enhancements via additional plug-in ROMs.

The HP-85 is an appliance computer: it was designed to be used as an application tool, not as a breadboard. This is the reason I haven't described the microprocessor (a custom-designed IC chip) or its unique architecture and other similar details. Hewlett-Packard has adopted the attitude of the mainframe computer manufacturers (not surprisingly, since they are one), and are more interested in showing the customer how to use the computer, not in describing the many features of its design (although that information is available to those who are interested).

Conclusion

The HP-85 combines many minicomputer features in one small package, features that either cost extra or are not available with other microcomputers. It is an impressive, well-built machine that reminds me of the engineering excellence of a Mercedes-Benz.

This machine will probably not affect the hobbyist market to any significant degree, primarily because of its price. Hewlett-Packard will sell a lot of these machines, but they will be found in labs and offices, not often in the home.■

✓See List of Advertisers on page 210

Clear Screen In the Blink of an Eye

By Roy D. Powers, Jr.

Since buying my OSI C1P BASIC-in-ROM computer, I have been irritated to have no easy, fast clear-screen routine. It is a lot of trouble to set up the USR (x) clear screen subroutine that OSI publishes, as well as a pain to type X = USR (x) for each use.

I knew there was little that could be done for BASIC-in-ROM machines, but felt that disk BASIC could be altered. When I expanded my C1P to minifloppy, I set to work on writing a program which would solve this problem.

After locating the command tables in the BASIC, I was disappointed to discover that they were full. It was then necessary to sacrifice an existing command and substitute the desired one.

After much consideration, I chose NULL, because it can easily be restored from the keyboard.

CLRS replaces the NULL command. The clear screen subroutine is poked into memory by the BASIC program, and is the same machine-language subroutine published by OSI. The subroutine has been modified to reside below the workspace. The workspace's lower limit has been raised to permit room for the subroutine. By placing the subroutine below the workshop, the program can be used without alteration for machines with different memory sizes. No buffer space was reserved, but the program is easily rewritten to allow their use.

With the subroutine loaded in the machine, all you must do to clear the screen is type CLRS and—presto—a fast clear screen. The command can be written into a program. For example

10 CLRS.

Placing the Program in BEXEC*

The easiest way to make CLRS a permanent command is to place the program in BEXEC* so that the subroutine is automatically entered each time on boot-up.

To do this, load BEXEC* and delete line number 11001. Enter the program (the NULL command in line 11008 is correct) and load the modified BEXEC* back on disk. Now when you boot up and type unlock, your screen will clear and CLRS is a permanent command.

```
11000 RESTORE:FORX=12927TO12950
11001 READS:POKEX,S:NEXT
11002 DATA169,32,160,08,162,0,157
11003 DATA0,208,232,208,250,238
11004 DATA135,50,136,208,244,169
11005 DATA 208,141,135,50,96
11006 POKE2158,127:POKE2159,50:POKE120,153:POKE121,50
11007 POKE709,67:POKE710,76:POKE711,82:POKE712,211
11008 NULL
11009 PRINT:PRINT"SYSTEM OPEN"
11010 NEW
```

CLRS program.

Reformatting Existing Programs

The workspace limits are automatically set by information contained on the disks' track format. Any new program that you write after booting up with the new BEXEC* will automatically be properly formatted. Your previous programs may be reformatted by doing the following:

1. Set up indirect files large enough to hold the program to be reformatted.
2. Load the program to be reformatted.
3. List into the indirect file.
4. Run BEXEC* (the BEXEC* must contain the CLRS routine).
5. Put the indirect file into the workspace.
6. Load the program back onto the disk.

If a program is executed which has not been reformatted, then BEXEC* must be reentered each time when the program is completed in order to restore CLRS. CLRS must not be used when running programs that have not been properly formatted.

Programs with specified addresses below the new workspace will not run. One example is "CREATE."

Other Uses of the CLRS Program

Booting up and selecting "Dir" instead of "Unlock" will not place the subroutine into memory. Also, running a program which has not been reformatted will destroy the subrou-

Address correspondence to Roy D. Powers, Jr., Mountain Empire Community College, Drawer 700, Big Stone Gap, VA 24219.

tine in memory. To overcome these problems, the program can be entered and stored in a separate file called CLRS. Running CLRS will place the subroutine in memory. You may elect to use this method instead of placing the program in BEXEC*.

Restoring NULL

NULL can be restored by poking 2158 and 2159 with 24 and 22, respectively. Remember that NULL has been replaced with CLRS, so the correct command will be CLRS 8 for eight NULLs. The use of CLRS alone will result in a syntax error.

Rewriting for Buffer Space

Memory locations 2158 and 2159 contain the low byte, high byte of the machine-language subroutine location. The routine can be moved to any unused location (note that some rewriting of the routine is necessary. Refer to your OSI manual for instructions on the clear screen routine).

Program Tested

This program has been tested and works on the C1, C3 and the C4. It will probably work on the C2 and C8 machines as well. ■


"I see Instant Software in your future..."
". . . .It's becoming very clear now . . . Your microcomputing life is going to be very exciting . . . Money! I see much money for you. Perhaps it is the money you will gain when Instant Software's new business applications guide your financial endeavors. I see travel . . . you will journey to distant worlds in distant times. You will build kingdoms from deserts and armies from slaves. Never will you be bored, adventurous one! Your programming burdens will lighten with new utilities—new tools. I see color! Many bright colors! I see new packaging for all these treasures . . . everything new! And . . . yes, a tall, handsome stranger who will guide you to these wonders. It will be your Instant Software dealer—a wise one indeed. All awaits you—all is yours for the asking. Soon . . . very soon!"

You're in good hands with this program that makes backup copies of machine-language, BASIC or assembler source tapes.

Tape Library Coverage

By Wayne L. Setzer

When my SWTP 6800 tape library started to grow, I wrote this program to make backups.

The program, written in assembly language (see Listing 1), had an unforeseen benefit. Since the delays caused by the BASIC interpreter when doing a save command are eliminated, I save close to a second per line. Loading time is thus cut in half. For example, one tape in my library takes four minutes and eight seconds to load at 1200 bits per second, while the copy takes only two minutes and three seconds.

The Program

The program is written to accept either of two types of tapes. First is the BASIC or assembler source tape. These always start with an STX character—a 02 (hexadecimal)—and end with an ETX character—a 03 (hexadecimal). The second type is the standard MIKBUG formatted tape. It begins with an S1 and ends with an S9.

The program has three main parts: read and store a tape into the buffer, write a new tape from the data stored in buffer and read the new tape and compare it to the data stored in the buffer. The program begins by asking for a command. It will always return to this routine after each command is completed. You can exit back to the MIKBUG monitor by typing a 4.

The read tape routine is used to read the tape you want to copy into a buffer. This routine also determines which type of tape you are reading and stores it for use in the write and compare routines.

The write-new-tape routine does just as it says. It is the routine that writes out to tape the data stored by the read routine. Before starting, the program will delay for five seconds after turning on the cassette in order to create a clean leader on the new tape. The write command also delays for five seconds at the end of the tape to create an ending leader.

The compare module reads the new tape that was copied by the write command and compares it with the data stored in the buffer by the read command of the original tape. If the tape does not compare, the program will report this. You can then go back and try the compare command again or rewrite the tape again.

Subroutines

The STCASR (start cassette for read), STPCAS (stop cassette and switch back to keyboard) and STCASW (start cassette for write) subroutines are used mainly to control my Percom CIS-30+ interface. This interface uses the RTS line to switch between keyboard or cassette for input. I also use these routines to control the number of stop bits. All tapes are written with two stop bits, but when reading a tape the ACIA only looks for one stop bit.

The tape input routine (INCHR) and tape output routine (OUTBYT) again were used for controlling the RTS line, since neither the MIKBUG nor the SWTBUG routine will allow this. Also I want to expand the program to cover binary-formatted tapes, and these routines will be needed.

The STRBYT (get start byte) is the routine which looks for the first valid character; if the COMMAND was a read, it will determine if the tape is type 1 or type 2. Type 1 is the BASIC or assembly source, and type 2 is the MIKBUG format tapes.

TSTEND, END1 and END2 are the routines which determine when the read or compare commands are complete.

Operation of the program is simple. The starting or entry address is 0100 (hexadecimal). The program will return to the screen the menu of commands, and you enter your choice. If you are just starting, enter a 1 for read tape. The program will then prompt you on the terminal. On the read command it will ask you to place a tape in the recorder and push play. When the tape is on the lead-in tone, hit -RET- (carriage return or enter). The read command will echo all data read from the tape to the terminal. The other commands will prompt you in the same manner.

This is my first complete program, and I'm sure there are more efficient ways of writing the routines. If you have any suggestions or questions, send an SASE and I'll try to answer them. ■

Wayne L. Setzer (3215 Dunaire Drive, Charlotte, NC 28205) is a field service technician on large computer systems.

Listing 1. 6800 Tape Copy and Compare program in assembly language.

```
                         NAM    TAPECOPY

                   ***************************************
                   *                                     *
                   *    6800 TAPE COPY AND COMPARE PROGRAM *
                   *                                     *
                   *    BY  WAYNE L SETZER               *
                   *          3215 DUNAIRE DR.           *
                   *          CHARLOTTE, NC 28205        *
                   *                                     *
                   ***************************************

                   ***************************************
                   *  THE START AND STOP SUBROUTINES ARE USED   *
                   *  TO SWITCH FROM KEYBOARD TO CASSETTE INPUT *
                   *  WHEN USING THE PERCOM CIS-30+ CONTROLLER. *
                   *  ALL TAPES ARE WRITTEN WITH 2 STOP BITS.   *
                   *  WHEN READING TAPES PROGRAM LOOKS FOR ONLY *
                   *  1 STOP BIT.                               *
                   ***************************************

                   ***************************************
                   *  THE PROGRAM WILL PROMPT FOR ALL INPUT.    *
                   *  THE -RET- ON MY TERMINAL IS THE SAME AS   *
                   *  CARRIAGE RETURN OR ENTER ON OTHER         *
                   *  TERMINALS.                                *
                   ***************************************

0100                     ORG    $0100
                         OPT    0,NOG
                   * EQUATES *
       E1AC        INEEE  EQU   $E1AC     INPUT ONE CHARACTER
       E07E        PDATA1 EQU   $E07E     OUTPUT A STRING OF DATA
       E0E3        CONTRL EQU   $E0E3     RETURN TO MONITOR
       8004        PORT1  EQU   $8004     PORT 1

                   * BEGINNING OF PROGRAM  *
                   * ROUTINE TO GET COMMAND *
0100 CE 0100       START  LDX   #START
0103 FF A048              STX   $A048
0106 8E A042              LDS   #$A042
0109 BD 02B2       GETCMD JSR   CLRTER
010C CE 0305              LDX   #HDGING
010F BD E07E              JSR   PDATA1
0112 BD E1AC              JSR   INEEE
0115 B7 02EC              STA A CMND
0118 81 31                CMP A #'1
011A 27 15                BEQ   RDTAPE
011C 81 32                CMP A #'2
011E 26 03                BNE   CK3
0120 7E 01E1              JMP   WRTAPE
0123 81 33         CK3    CMP A #'3
0125 26 03                BNE   CK4
0127 7E 0191              JMP   CPTAPE
012A 81 34         CK4    CMP A #'4
012C 26 DB                BNE   GETCMD
012E 7E E0E3              JMP   CONTRL
                   * ROUTINE TO READ AND STORE TAPE *
0131 BD 02B2       RDTAPE JSR   CLRTER
0134 CE 03D9              LDX   #MSG1
0137 BD E07E              JSR   PDATA1
013A BD 02A4              JSR   GETCR
013D BD 0277              JSR   STCASR
0140 CE 0603              LDX   #TPBUFF
0143 BD 0231       RD2    JSR   STRBYT
0146 A7 00                STA A X
0148 08            NXTBYT INX
0149 BD 02B9              JSR   INCHR
014C A7 00                STA A X
014E BD 016C              JSR   TSTEND
0151 B6 02ED              LDA A TAPETY
0154 81 31                CMP A #'1
0156 27 0A                BEQ   RD4
0158 C1 00                CMP B #0
015A 27 EC                BEQ   NXTBYT
```

More →

✓See List of Advertisers on page 210

Listing continued.

```
015C B6 02EE         LDA A   HLDBYT
015F 08              INX
0160 A7 00           STA A   X
0162 C1 45   RD4     CMP B   #'E
0164 26 E2           BNE     NXTBYT
0166 CE 056C         LDX     #RDCOMP
0169 7E 0298         JMP     ENDIT
             * TEST FOR END ROUTINE *
016C F6 02ED TSTEND  LDA B   TAPETY
016F C1 31           CMP B   #'1
0171 27 02           BEQ     END1
0173 20 07           BRA     END2
             * BASIC OR ASSEMBLER SOURCE END *
0175 81 03   END1    CMP A   #3
0177 26 17           BNE     RET
0179 C6 45           LDA B   #'E
017B 39              RTS
             * MIKBUG FORMATE END *
017C C6 00   END2    LDA B   #0
017E 81 53           CMP A   #'S
0180 26 0E           BNE     RET
0182 C6 46           LDA B   #'F
0184 BD 02B9         JSR     INCHR
0187 B7 02EE         STA A   HLDBYT
018A 81 39           CMP A   #'9
018C 26 02           BNE     RET
018E C6 45           LDA B   #'E
0190 39      RET     RTS
             * ROUTINE TO READ AND COMPARE NEW TAPE *
0191 BD 02B2 CPTAPE  JSR     CLRTER
0194 CE 042A         LDX     #MSG2
0197 BD E07E         JSR     PDATA1
019A BD 02A4         JSR     GETCR
019D BD 0277         JSR     STCASR
01A0 CE 0603         LDX     #TPBUFF
01A3 BD 0231         JSR     STRBYT
01A6 A1 00   CP1     CMP A   X
01A8 26 28           BNE     ERRCAS
01AA 08      CP2     INX
01AB BD 02B9         JSR     INCHR
01AE A1 00           CMP A   X
01B0 26 20           BNE     ERRCAS
01B2 BD 016C         JSR     TSTEND
01B5 B6 02ED         LDA A   TAPETY
01B8 81 31           CMP A   #'1
01BA 27 0C           BEQ     CP3
01BC C1 00           CMP B   #0
01BE 27 EA           BEQ     CP2
01C0 08              INX
01C1 B6 02EE         LDA A   HLDBYT
01C4 A1 00           CMP A   X
01C6 26 0A           BNE     ERRCAS
01C8 C1 45   CP3     CMP B   #'E
01CA 26 DE           BNE     CP2
01CC CE 059D         LDX     #CMCOMP
01CF 7E 0298         JMP     ENDIT
01D2 BD 0283 ERRCAS  JSR     STPCAS
01D5 CE 0532         LDX     #TAPERR
01D8 BD E07E         JSR     PDATA1
01DB BD E1AC         JSR     INEEE
01DE 7E 0109         JMP     GETCMD
             * ROUTINE TO WRITE A NEW TAPE *
01E1 BD 02B2 WRTAPE  JSR     CLRTER
01E4 CE 048A         LDX     #MSG3
01E7 BD E07E         JSR     PDATA1
01EA BD 02A4         JSR     GETCR
01ED BD 026B         JSR     STCASW
01F0 BD 02E1         JSR     DELAY
01F3 CE 02F3         LDX     #WROUT
01F6 A6 00   WR1     LDA A   X
01F8 BD 02CF         JSR     OUTBYT
01FB 81 00           CMP A   #0
01FD 27 03           BEQ     WR2
01FF 08              INX
0200 20 F4           BRA     WR1
0202 CE 0603 WR2     LDX     #TPBUFF
```

More

4MHZ, DOUBLE DENSITY, COLOR & B/W GRAPHICS . . THE LNW80 COMPUTER

When you've compared the features of an LNW80 Computer, you'll quickly understand why the LNW80 is the ultimate TRS80 software compatible system. LNW RESEARCH offers the most complete microcomputer system at an outstanding low price.
We back up our product with an unconventional 6 month warranty and a 10 days full refund policy, less shipping charges.

LNW80 Computer $1,450.00
LNW80 Computer w/B&W Monitor & one 5" Drive $1,914.00
All orders must be prepaid, CA residents please include 6% sales tax.
Contact us for shipping charges

* TRS80 Product of Tandy Corporation.
** PMC Product of Personal Microcomputer, Inc.

COMPARE THE FEATURES AND PERFORMANCE

FEATURES	LNW80	PMC-80**	TRS-80* MODEL III
PROCESSOR	4.0 MHZ	1,8 MHZ	2.0 MHZ
LEVEL II BASIC INTERP.	YES	YES	LEVEL III BASIC
TRS80 MODEL 1 LEVEL II COMPATIBLE	YES	YES	NO
48K BYTES RAM	YES	YES	YES
CASSETTE BAUD RATE	500/1000	500	500/1500
FLOPPY DISK CONTROLLER	SINGLE/ DOUBLE	SINGLE	SINGLE/ DOUBLE
SERIAL RS232 PORT	YES	YES	YES
PRINTER PORT	YES	YES	YES
REAL TIME CLOCK	YES	YES	YES
24 X 80 CHARACTERS	YES	NO	NO
VIDEO MONITOR	YES	YES	YES
UPPER AND LOWER CASE	YES	OPTIONAL	YES
REVERSE VIDEO	YES	NO	NO
KEYBOARD	63 KEY	53 KEY	53 KEY
NUMERIC KEY PAD	YES	NO	YES
B/W GRAPHICS, 128 X 48	YES	YES	YES
HI-RESOLUTION B/W GRAPHICS, 480 X 192	YES	NO	NO
HI-RESOLUTION COLOR GRAPHICS (NTSC), 128 X 192 IN 8 COLORS	YES	NO	NO
HI-RESOLUTION COLOR GRAPHICS (RGB), 384 X 192 IN 8 COLORS	OPTIONAL	NO	NO
WARRANTY	6 MONTHS	90 DAYS	90 DAYS
TOTAL SYSTEM PRICE	$1,914.00	$1,840.00	$2,187.00
LESS MONITOR AND DISK DRIVE	$1,450.00	$1,375.00	---

LNW80

- BARE PRINTED CIRCUIT BOARD & MANUAL $89.95

The LNW80 - A high-speed color computer totally compatible with the TRS-80*. The LNW80 gives you the edge in satisfying your computation needs in business, scientific and personal computation. With performance of 4 MHz, Z80A CPU, you'll achieve performance of over twice the processing speed of a TRS-80*. This means you'll get the performance that is comparable to the most expensive microcomputer with the compatibility to the world's most popular computer (TRS-80*) resulting in the widest software base.

FEATURES:
- TRS-80 Model 1 Level II Software Compatible
- High Resolution Graphics
  - RGB Output - 384 x 192 in 8 Colors
  - NTSC Video or RF MOD - 128 x 192 in 8 Colors
  - Black and White - 480 x 192
- 4 MHz CPU
- 500/1000 Baud Cassette
- Upper and Lower Case
- 16K Bytes RAM, 12K Bytes ROM
- Solder Masked and Silkscreened

LNW SYSTEM EXPANSION

- *BARE PRINTED CIRCUIT BOARD AND MANUAL* $69.95
WITH GOLD CONNECTORS $84.95

The System Expansion will allow you to expand your LNW80, TRS-80*, or PMC-80** to a complete computer system that is still totally software compatible with the TRS-80* Model 1 Level II.

FEATURES:
- 32K Bytes Memory
- 5" Floppy Controller
- Serial RS232 20ma I/O
- Parallel Printer
- Real Time Clock
- Screen Printer Bus
- On Board Power Supply
- Solder Masked and Silkscreened

LNDoubler & DOS PLUS 3.3D

- Assembled and Tested W/DOS PLUS 3.3D. $175.00

Double-density disk storage for the LNW Research's "System Expansion" or the Tandy's "Expansion Interface". The LNDoubler™ is totally software compatible with any double density software generated for the Percom's Doubler***. The LNDoubler™ provides the following outstanding features.

- Store up to 350K bytes on a single 5" disk
- Single and double density data separation
- Precision write precompensation circuit
- Software switch between single and double density
- Easy plug in installation requiring no etch cuts, jumpers or soldering
- 35, 40, 77, 80 track 5" disk operation
- 120 day parts and labor Warranty

*** Doubler is a product of Percom Data Company, Inc.

DOS PLUS 3.3D

Micro Systems software's double density disk operating system. This operating system contains all the outstanding features of a well developed DOS, with ease in useability.

KEYBOARD

LNW80 KEYBOARD KIT $84.95

The Keyboard Kit contains a 63 key plus a 10 key, P.C. board, and remaining components.

CASE

LNW80 CASE *$84.95*

The streamline design of this metal case will house the LNW80, LWN System Expansion, LNW80 Keyboard, power supply and fan, LNDoubler™, or LNW Data Separator. This kit includes all the hardware to mount all of the above. Add $12.00 for shipping

PARTS AVAILABLE FROM LNW RESEARCH
- 4116 - 200ns RAM
  - 6 chip set $26.00
  - 8 chip set $33.50
  - 16 chip set $64.00
  - 24 chip set $94.00
  - 32 chip set $124.00

- LNW80 "Start up parts set" LNW80-1 . . . . $82.00
- LNW80 "Video parts set" LNW80-2 . . . . $31.00
- LNW80 Transformer LNW80-3 . . . . $18.00
- LNW80 Keyboard cable LNW80-4 . . . . $16.00
- 40 Pin computer to expansion cable $15.00
- System Expansion Transformer $19.00
- Floppy Controller (FD1771) and UART (TR1602) . . . $30.00

LNW RESEARCH CORPORATION ✓198

2620 WALNUT
TUSTIN CA. 92680

ORDERS & INFO. NO. 714-544-5744
SERVICE NO. 714-641-8850

VISA MASTER CHARGE ACCEPTED

UNLESS NOTED ADD $3 FOR SHIPPING

✓See List of Advertisers on page 210

Listing continued.

```
0205 A6 00     WR3    LDA A  X
0207 BD 02CF          JSR    OUTBYT
020A F6 02ED          LDA B  TAPETY
020D C1 31            CMP B  #'1
020F 26 07            BNE    WR5
0211 81 03            CMP A  #3
0213 27 13            BEQ    ENDWR
0215 08        WR4    INX
0216 20 ED            BRA    WR3
0218 81 53     WR5    CMP A  #'S
021A 26 F9            BNE    WR4
021C 08               INX
021D A6 00            LDA A  X
021F BD 02CF          JSR    OUTBYT
0222 81 39            CMP A  #'9
0224 26 EF            BNE    WR4
0226 20 00            BRA    ENDWR
0228 BD 02E1   ENDWR  JSR    DELAY
022B CE 05D1          LDX    #WRCOMP
022E 7E 0298          JMP    ENDIT
                      * GET START BYTE *
0231 BD 02B9   STRBYT JSR    INCHR
0234 81 02            CMP A  #$2
0236 26 06            BNE    STRT2
0238 C6 31            LDA B  #'1
023A F7 02ED          STA B  TAPETY
023D 39               RTS
023E 81 53     STRT2  CMP A  #'S
0240 26 EF            BNE    STRBYT
0242 B7 02EE          STA A  HLDBYT
0245 BD 02B9          JSR    INCHR
0248 81 31            CMP A  #'1
024A 26 E5            BNE    STRBYT
024C F6 02EC          LDA B  CMND
024F C1 33            CMP B  #'3
0251 27 0C            BEQ    STRT3
0253 F6 02EE          LDA B  HLDBYT
0256 E7 00            STA B  X
0258 08               INX
0259 C6 32            LDA B  #'2
025B F7 02ED          STA B  TAPETY
025E 39               RTS
025F F6 02EE   STRT3  LDA B  HLDBYT
0262 E1 00            CMP B  X
0264 26 02            BNE    STRT4
0266 08               INX
0267 39               RTS
0268 7E 01D2   STRT4  JMP    ERRCAS
                      * START CASSETTE FOR WRITE *
026B CE 8004   STCASW LDX    #PORT1
026E C6 13            LDA B  #$13     RESET ACIA
0270 E7 00            STA B  X
0272 C6 11            LDA B  #$11     RTS=0, 2 STOP BITS
0274 E7 00            STA B  X
0276 39               RTS
                      * SWITCH TO CASSETTE FOR READ *
0277 CE 8004   STCASR LDX    #PORT1
027A C6 43            LDA B  #$43     RESET ACIA
027C E7 00            STA B  X
027E C6 55            LDA B  #$55     RTS=1, 1 STOP BIT
0280 E7 00            STA B  X
0282 39               RTS
                      * STOP CASSETTE, SWITCH BACK TO KEYBOARD *
0283 FF A012   STPCAS STX    $A012
0286 CE 8004          LDX    #PORT1
0289 C6 13            LDA B  #$13     RESET ACIA
028B E7 00            STA B  X
028D C6 15            LDA B  #$15     RTS=0, 1 STOP BIT
028F E7 00            STA B  X
0291 BD 02B2          JSR    CLRTER
0294 FE A012          LDX    $A012
0297 39               RTS
                      * ENDIT ROUTINE *
0298 BD 0283   ENDIT  JSR    STPCAS
029B BD E07E          JSR    PDATA1
```

More

Listing continued.

```
029E BD E1AC        JSR    INEEE
02A1 7E 0109        JMP    GETCMD
                    * WAIT FOR C/R INPUT *
02A4 BD E1AC GETCR  JSR    INEEE
02A7 81 0D          CMP A  #$D
02A9 26 F9          BNE    GETCR
02AB CE 02FA        LDX    #STRIT
02AE BD E07E        JSR    PDATA1
02B1 39             RTS
                    * CLEAR TERMINAL SCREEN *
02B2 CE 02EF CLRTER LDX    #CLRSCR
02B5 BD E07E        JSR    PDATA1
02B8 39             RTS
                    * TAPE INPUT ROUTINE FOR MP-S (ACIA) INTERFACE *
02B9 FF A012 INCHR  STX    $A012
02BC 37             PSH B
02BD CE 8004        LDX    #PORT1
02C0 E6 00   IN1    LDA B  0,X
02C2 57             ASR B
02C3 24 FB          BCC    IN1
02C5 A6 01          LDA A  1,X
02C7 BD 02D5        JSR    OUT1      ECHO DATA TO TERMINAL
02CA FE A012        LDX    $A012
02CD 33             PUL B
02CE 39             RTS
                    * TAPE OUTPUT ROUTINE FOR MP-S (ACIA) INTERFACE *
02CF FF A012 OUTBYT STX    $A012
02D2 CE 8004        LDX    #PORT1
02D5 E6 00   OUT1   LDA B  0,X
02D7 57             ASR B
02D8 57             ASR B
02D9 24 FA          BCC    OUT1
02DB A7 01          STA A  1,X
02DD FE A012        LDX    $A012
02E0 39             RTS
                    * 5 SECOND TIME DELAY *
02E1 CE 0C00 DELAY  LDX    #$0C00    SET FOR 5 SECONDS
02E4 4F             CLR A
02E5 4C      TD1    INC A            TIME DELAY LOOP
02E6 26 FD          BNE    TD1
02E8 09             DEX
02E9 26 FA          BNE    TD1
02EB 39             RTS
                    * VARIABLES *
02EC 0001    CMND   RMB    1         STORE TYPE OF COMMAND
02ED 0001    TAPETY RMB    1         STORE TYPE OF TAPE
02EE 0001    HLDBYT RMB    1         HOLD A BYTE
                    * CONSTANTS *
02EF 10      CLRSCR FCB    $10,$16,,$4
02F3 FF      WROUT  FCB    $FF,$FF,$FF,$FF,$D,$A,$0
02FA 53      STRIT  FCC    /STARTING/
0302 0D             FCB    $D,$A,$4
0305 20      HDGING FCC    /                              /
0324 36             FCC    /6 8 0 0/
032B 0D             FCB    $D,$A,$A
032E 20             FCC    /                   TAPE COPY /
034C 41             FCC    /AND COMPARE PROGRAM/
035F 0D             FCB    $D,$A,$A,$A
0363 31             FCC    /1.  READ TAPE TO BE COPIED/
037D 0D             FCB    $D,$A
037F 32             FCC    /2.  WRITE NEW TAPE/
0391 0D             FCB    $D,$A
0393 33             FCC    /3.  COMPARE NEW TAPE TO OLD TAPE/
03B3 0D             FCB    $D,$A
03B5 34             FCC    /4.  EXIT TO MONITOR/
03C8 0D             FCB    $D,$A,$A
03CB 45             FCC    /ENTER OPTION /
03D8 04             FCB    $4
03D9 50      MSG1   FCC    /PLACE TAPE IN RECORDER /
03F0 41             FCC    /AND PUSH PLAY./
03FE 0D             FCB    $D,$A
0400 57             FCC    /WHEN TAPE IS ON LEAD-IN /
0418 54             FCC    /TONE HIT -RET-./
0427 0D             FCB    $D,$A,$4
042A 50      MSG2   FCC    /PLACE TAPE TO BE COMPARED /
0444 49             FCC    /IN RECORDER AND PUSH PLAY./
```

More

✓See List of Advertisers on page 210

Listing continued.

```
0045E 0D              FCB   $D,$A
0460 57               FCC   /WHEN TAPE IS ON LEAD-IN /
0478 54               FCC   /TONE HIT -RET-./
0487 0D               FCB   $D,$A,$4
048A 50        MSG3   FCC   /PLACE BLANK TAPE IN RECORDER /
04A7 41               FCC   /AND SET UP TO RECORD./
04BC 0D               FCB   $D,$A
04BE 41               FCC   /AFTER HITTING -RET- ON KEYBOARD /
04DE 50               FCC   /PROGRAM WILL DELAY 5 SECONDS,/
04FB 0D               FCB   $D,$A
04FD 49               FCC   /IN ORDER TO CREATE A LEAD-IN /
051A 54               FCC   /TONE ON THE NEW TAPE./
052F 0D               FCB   $D,$A,$4
0532 0D        TAPERR FCB   $D,$A,$A,$A,$A
0537 54               FCC   /TAPE DOES NOT COMPARE!!!!/
0550 0D               FCB   $D,$A,$A
0553 48               FCC   /HIT -RET- TO RESTART./
0569 0D               FCB   $D,$A,$4
056C 54        RDCOMP FCC   /TAPE READ NOW COMPLETE. /
0584 48               FCC   /HIT -RET- TO CONTINUE./
059A 0D               FCB   $D,$A,$4
059D 54        CMCOMP FCC   /TAPE COMPARE NOW COMPLETE. /
05B8 48               FCC   /HIT -RET- TO CONTINUE./
05CE 0D               FCB   $D,$A,$4
05D1 54        WRCOMP FCC   /TAPE WRITE NOW COMPLETE. /
05EA 48               FCC   /HIT -RET- TO CONTINUE./
0600 0D               FCB   $D,$A,$4
0603 0001      TPBUFF RMB   1        TAPE INPUT BUFFER

A048                  ORG   $A048
A048 0100             FDB   START
                      END
```

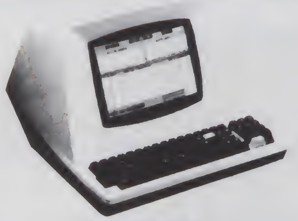
✓See List of Advertisers on page 210

AUTOMATED EQUIPMENT INCORPORATED

96

18430 Ward, Fountain Valley, CA 92708

Call Toll-Free for latest low prices!
1-800-854-7635 Outside CA
714-963-1414 Inside CA
1-714-963-1414 Outside Cont. USA
Prices change daily to meet competition.



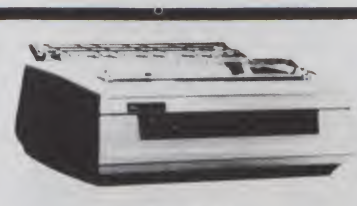
TERMINALS

TELEVIDEO 912 B	CALL
TELEVIDEO 912 C	CALL
TELEVIDEO 920 B	CALL
TELEVIDEO 920 C	CALL
SOROC IQ-120	675
ZENITH A-19	750
HAZELTINE 1500	995
TELEVIDEO 950	CALL

NORTHSTAR

HRZ-1D-32K	CALL
HRZ-2D-32K	CALL
HRZ-1Q-32K	CALL
HRZ-2Q-32K	CALL
ADDITIONAL 16K RAM	CALL
ADDITIONAL 32K RAM	CALL
HARD DISK SYSTEM	CALL

SPECIAL MEMORY CONFIGURATIONS AVAILABLE, INCLUDING SYSTEMS GROUP.



PRINTERS

NEC 5510 (TRACT., RIB., THIM)	2600
NEC 5515 (TRACT., RIB., THIM)	2600
NEC 5520 (TRACT., RIB., THIM)	2900
NEC 5530 (TRACT., RIB., THIM)	2540
TI-810	1495
TI-820	1625
EPSON MX 80	CALL
EPSON MX 70	CALL
EPSON MX 100	CALL
ANADEX DP-9500/9501	CALL

VIP's customarily need top quality and service within stringent budgetary limits. naturally, they call A.E.I.

Item: A.E.I. tests virtually every* item sold, before shipping.

Item: A.E.I. initializes every piece of software, and makes a copy. If you have a problem, A.E.I. can check to locate the problem by using the in-house duplicate, saving you time.

Item: A.E.I. has in-house expetise on every item offered.

Item: A.E.I. prices are competitive with the lowest, no-service, no in-house expertise dealer.

40% OF ALL A.E.I. SALES ARE TO PUBLIC AND SEMI-PUBLIC INSTITUTIONS.

A partial list** of A.E.I. customers who purchased during the last quarter of 1980 includes: University of Nebraska, University of Virginia, U.S. Dept. of Interior, University of Kentucky, University of California, Massachusetts Institute of Technology, U.S. Air Force, Naval Air Development Center, Brown University, University of Oregon, University of Utah, Ohio State University, University of Southern California, California State Dept. of Water Resources, University of Michigan, and Princeton University.

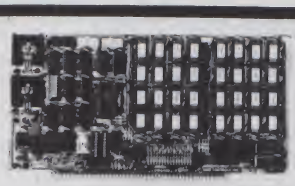
In the private sector, an equal number of major national organizations purchased during the same period.

You can call A.E.I. too!

VIP's Call A.E.I.

*With our normal fast shipping, we have time to test; expedited shipment may preclude testing.

**A.E.I. does not wish to imply that any of these fine organizations endorse A.E.I., merely that A.E.I. is proud to have them as customers.



S.S.M.	KIT	ASM
CB2 Z80 CPU	$200	$265
VB1C VIDEO	140	190
VB2 VIDEO	155	210
VB3 VIDEO	375	440
IO4 INTERFACE	165	225
SB1 SYNTHESIZER	195	270

MEASUREMENT SYSTEMS MEMORY

DM3200 32K 4MHZ	480
DM6400 64K 4MHZ	595
DMB3200 32K 4MHZ BANK SELECT	630
DMB6400 64K 4MHZ BANK SELECT	745

DYNABYTE SYSTEMS ALSO AVAILABLE—CALL FOR THE EXCITING NEWS ABOUT NEW COMPETITIVE PRICING!

WE WILL TRY TO BEAT ANY ADVERTISED PRICE.

TERMS: All prices listed are cash discounted and subject to change. COD's are 2% additional. Universities and well-rated firms Net-10.

SHIPPING: ADDITIONAL IN ALL CASES.

TECHNICAL ASSISTANCE: WE TRY TO HELP INTERFACE AND TROUBLESHOOT—CALL (714) 963-1414.

Open Monday-Friday, 8:00 a.m.-5:00 p.m.

Dealer Inquiries Invited.

MORROW

DECISION 1 BASIC	$1550
DISCUS-1 2 DRIVE	1550
DISCUS 2D 1 DRIVE	935
DISCUS 2D 2 DRIVE	1550
DISCUS 2+2 1 DRIVE	1250
DISCUS 2+2 2 DRIVE	2190
DISC JOCKEY 2D	350
DISCUS M26 HARD DISC	3925
ADDITIONAL HARD DISC	3685
DISCUS M10	2990

MISCELLANEOUS

NEC THIMBLES	$16
NEC RIBBONS	7
RS2332 CABLE 5'	20
RS 232 CABLE 10"	25
LEDEX MONITOR	145
NOVATION CAT	145

SOFTWARE

WORDSTAR	$330
MAILMERGE	100
DATASTAR	250
N.S. PASCAL	160
MAGIC WAND	290
C BASIC	100

GRAHAM-DORIAN

JOB COSTING	$700
INVENTORY	475
CASH REG.	475
APARTMENT	475
MEDICAL	700

ADDITIONAL SOFTWARE DISCOUNTS WITH SYSTEM PURCHASE

DISCS—BOX OF 10

VERBATUM 5¼ 1 SIDE	$27
VERBATUM 5¼ 2 SIDE	45
VERBATUM 8 1 SIDE	35
VERBATUM 8 2 SIDE	55
OTHERS	CALL
PLASTIC STORAGE BOXES	3

OUR SYSTEMS WORK!

All systems normally tested and configured in our repair facility before delivery. Service contracts available. Prompt repairs and warranties.

Here's a useful reference for the Z-80 programmer: a complete description of the Z-80 instruction set and machine code.

The Z-80 Condensed

By R. Daniel Bishop

Writing an assembly-language program and translating it into machine-language code can be a formidable task, even when the program is to perform a relatively simple function. The Z-80 microprocessor adds to the challenge by providing a more elaborate instruction set than some of the other popular microprocessors.

The difficulty lies in trying to keep track of every item in the instruction set so that the Z-80 can be used to best advantage. I decided to set about summarizing the instruction set, leaving nothing out, but condensing the list to a single page or two which could be searched at a glance for the desired instruction. This job looked awesome at first, since the TRS-80 Editor Assembler manual devotes 112 pages to the Z-80 instruction set.

As I began work on the project, I realized that if I could include in my summary the hex machine code corresponding to each of the Z-80 instructions and the relevant flag conditions resulting from each operation, I could avoid using the Editor Assembler program when writing simple programs. Although the Editor Assembler from Radio Shack allows the entry of assembly-language mnemonics directly from the keyboard, the method for testing and debugging programs is very cumbersome.

First, the assembly-language program is typed out and stored in the editor's text buffer. Then it can be assembled, at which time some coding errors are flagged. When these have been corrected and the revised program reassembled, the program must be saved on cassette tape, and subsequently reloaded from tape as a SYSTEM program before it can be executed. If alterations are needed, it is necessary to go back to the Editor Assembler program and continue working on the program, involving still more tape loads and saves. Thus, for shorter programs it is easier to copy down the hex object code displayed on the screen at the time of assembly, and then use a system monitor such as Radio Shack's TBUG or Small System Software's RSM-2 to enter the program into RAM one byte at a time.

One advantage of RSM-2 over TBUG is the ability to display memory in assembly-language mnemonics rather than just as hex code. Once a program has been entered by hand into RAM, using the RSM-2 edit memory (E) feature, it can be displayed in assembly-language mnemonics with the (S) command. This makes it much easier to catch coding errors.

If both the assembly-language mnemonics and the corresponding machine-language (hexadecimal) code

Bit:	0	1	2	3	4	5	6	7
Condition:	NZ	Z	NC	C	P/V odd	P/V even	S pos P	S neg M
Register:	B	C	D	E	H	L	(HL)*	A
b1,c1,r1	0	0	1	1	2	2	3	3
b2,c2,r2	0	8	0	8	0	8	0	8
r3	0	1	2	3	4	5	6	7

* (IX + d) and (IY + d) have the same code as (HL), with DD or FD (respectively) preceding the (HL) code, and the hex code for the displacement *d* following the (HL) code.

JR jumps can be conditional based only on Z or C flags.

Table 1. Values to be used in determining hex machine codes for instructions involving specific registers, conditions or bits. This table is to be used with Tables 4–9 to compute hex machine codes.

Dr. Dan Bishop is a microcomputer consultant and owner of Custom Comp, PO Box 429, Buena Vista, CO 81211.

Register Pair:	BC	DE	HL	SP
rr	0	1	2	3

Table 2. Values to be used in determining hex machine codes for 16-bit instructions involving register pairs. IX and IY may substitute for HL in this table. This table is to be used with Tables 4–9 to compute hex machine codes.

could be included in my summary, I could write the assembly program, convert the mnemonics into hex code immediately from the tables, and then use RSM-2 to enter the code and list the mnemonics for verification. Since RSM-2 is high in the address space, I would then be able to test the program, edit it and retest it as often as necessary without once having to work with the cassette recorder until the project was finished and the final version of the program was ready to be saved. Furthermore, since my version of RSM-2 is on disk with Radio Shack's Tapedisk utility program, which allows a disk save of any specified area of RAM, I can avoid the use of the cassette tape recorder entirely.

The Code is the Key

There are 694 separate Z-80 instructions, and I had some doubts about condensing the instruction set when I began studying the codes in detail. But soon the pattern began to emerge from the morass of hex codes. I found that, with a few simple rules and the ten tables accompanying this article, the entire Z-80 instruction set becomes more manageable. The ten tables include not only the instruction set mnemonics and hex codes, but also the flag condition codes and the number of machine cycles for each instruction as well.

The major difference between my method of representing the hex codes and the representations found in the Mostek and Radio Shack manuals is the presentation of the codes in hexadecimal digit form. This makes the job of determining hex codes much simpler. The manuals present the LD A,D instruction as (01 rrr 'rrr'), where rrr is 111 for A and 'rrr' is 010 for D. This binary representation must then be split into two parts and each half evaluated to obtain a hex digit.

Using Tables 4–9 in this article, the two-digit hex code can be figured directly from the hex code representation in the table and the keys to those

Code	Flag	Condition (Set condition given; reset if condition not met)
X	All	Flag condition not affected by operation.
U	All	Flag condition unknown.
0	All	Flag is reset.
1	All	Flag is set.
S-2	S	Set if result is negative.
S-3	S	Set if A is negative after operation.
Z-2	Z	Set if result is zero.
Z-3	Z	Set if A is zero after operation.
Z-4	Z	Set if specified bit is zero.
Z-5	Z	Set if B – 1 = 0.
P-2	P/V	Set if overflow.
P-3	P/V	Set if parity is even.
P-4	P/V	Set if BC – 1≠0.
P-5	P/V	Set if r (or A) was 7FH before operation.
P-6	P/V	Set if r (or A) was 80H before operation.
P-7	P/V	Set if parity of A is even after operation.
C-2	C	Set if carry from bit 7.
C-3	C	Set if borrow.
C-4	C	Set if carry from bit 15.
C-5	C	Set if accumulator was not 00H before operation.
C-6	C	Set if carry flag was 0 before operation.
C-7	C	Data from bit 7 of the accumulator or source register.
C-8	C	Data from bit 0 of the accumulator or source register.
H-2	H	Set if carry from bit 3.
H-3	H	Set if no borrow from bit 4.
H-4	H	Set if carry from bit 11.
H-5	H	Set if no borrow from bit 12.

F register

	S	Z	x	H	x	P	N	C
bit:	7	6	5	4	3	2	1	0

Table 3. Explanation of codes used to describe flag conditions following specified operations from the following tables. The flag codes appear between brackets, [ ], in the following order: S (sign), Z (zero), H (half carry), P/V (parity-overflow), N (add-subtract) and C (carry). This order corresponds to the bit location in the F register.

8-Bit Load Group:

LD	r,r'	(r1 + 4)(r2 + r'3)			
LD	r,n	(r1)(r2 + 6) nn			
LD	A,(BC)	0A	LD	(BC),A	02
LD	A,(DE)	1A	LD	(DE),A	12
LD	A,(nn)	3A nn nn	LD	(nn),A	32 nn nn
LD	A,I	ED 57	LD	I,A	ED 47
LD	A,R	ED 5F	LD	R,A	ED 4F

16-Bit Load Group:

LD	rr,(nn)	ED (rr + 4)B nn nn			
LD	(nn),rr	ED (rr + 4)3 nn nn			
LD	rr,nn	(rr)1 nn nn			
LD	HL,(nn)	2A nn nn			
LD	(nn),HL	22 nn nn			
LD	SP,HL	F9	PUSH	rr	(rr + 12)5
			POP	rr	(rr + 12)1

Exchange Group:

EX	DE,HL	EB	EX	AF,AF'	08	EX	(SP),HL	E3	EXX D9

Jump Group:

CALL	nn	CD nn nn	RET		C9	JP	nn	C3 nn nn	
CALL	cc,nn	(c1 + 12)(c2 + 4) nn nn	RET	cc	(c1 + 12)(c2)	JP	cc,nn	(c1 + 12)(c2 + 2) nn nn	
						JP	(HL)	E9	
JR	d	18 dd	JR	cc,d	(c1 + 2)(c2) dd	DJNZ	d	10 dd	B is decremented; jump if B≠0.

1. In this table, only the 8-bit load group uses a displacement with IX and IY. The other cases substitute IX or IY directly for HL.
2. (HL), (IX + d) and (IY + d) can appear only once in an expression and cannot be mixed.
3. Where two instructions are possible for HL, one using rr, the other explicitly stating HL, only the latter may be used with HL, IX or IY.
4. The EX DE,HL instruction applies only to HL, and not IX and IY.
5. For PUSH and POP, substitute register pair AF for SP in Table 2.
6. JR instructions can be conditional only on Z and C flags, not on P/V or S.
7. "d" is a displacement value between −128 and +127 that is added to the PC register if the conditions are met, making the jump relative to the current program position.

Table 4. Z-80 instruction groups that have no effect on the flag registers.

Block Transfer Group:

LDI	ED A0	[xx040x] (DE)←(HL); DE and HL increment; BC decrements.	LDD	ED A8	[xx040x] (DE)←(HL); DE, HL and BC decrement.
LDIR	ED B0	[xx000x] As LDI, but repeats until BC = 0.	LDDR	ED B8	[xx000x] As LDD, but repeats until BC = 0.

Search Group: [22341x] No change occurs in the register or memory values.

CPI	ED A1	A – (HL) is performed to set flags. HL increments; BC decrements.	CPD	ED A9	A – (HL) is performed to set flags. Both HL and BC decrement.
CPIR	ED B1	As CPI, but repeats until BC = 0 or A = (HL).	CPDR	ED B9	As CPD, but repeats until BC = 0 or A = (HL).

Logic Group: [221300] Operation carried out with A; results stored in A.

AND	r	A(r3)	AND	n	E6 nn	OR	r	B(r3)	OR	n	F6 nn	XOR	r	A(r3 + 8)	XOR	n	EE nn

8-Bit Arithmetic Group: Operation always with A; results stored in A.

ADD	A,r	8(r3)	[222202]	ADD	A,n	C6 nn	[222202]
ADC	A,r	8(r3 + 8)	[222202] Carry flag is added to the sum of A and r.	ADC	A,n	CE nn	[222202] Carry flag is added to the sum of A and n.
SUB	r	9(r3)	[223213]	SUB	n	D6 nn	[223213]
SBC	A,r	9(r3 + 8)	[223213] Carry flag is subtracted from result of A – r.	SBC	A,n	DE nn	[223213] Carry flag is subtracted from result of A – n.
INC	r	(r1)(r2 + 4)	[22250x]	DEC	r	(r1)(r2 + 5)	[22361x]
CP	r	B(r3 + 8)	[223213] A – r is performed to set flags. No change in A or r.	CP	n	FE nn	[223213] A – n is performed to set flags. A is unchanged.

16-Bit Arithmetic Group:

ADD	HL,rr	(rr)9	[xx4x04]	INC	rr	(rr)3	[xxxxxx]	DEC	rr	(rr)B	[xxxxxx]
ADC	HL,rr	ED (rr + 4)A	[224204] Carry flag is also added to sum.	SBC	HL,rr	ED (rr + 4)2	[225213] Carry flag is subtracted from result.				

Table 5. Z-80 instructions involving arithmetic processing. Only the eight-bit arithmetic group and the logic group allow (IX + d) and (IY + d) to substitute for (HL). The 16-bit arithmetic group allows IX or IY to substitute for HL only in ADD, INC and DEC instructions. Note that HL, IX and IY cannot be mixed in the same instructions.

Ω WEST COAST / EAST COAST Ω

Special of the Month!

EPSON MX-80

INTERFACES: IEEE $55, TRS-80 $35, APPLE INTERFACE & CABLE $90, RS-232 $70

EPSON MX-80FT $599
MX-100FT $769

No Surcharge for Credit Cards Orders • We Accept C.O.D.'s
All Equipment Factory Fresh With MFT. Warranty • Stock Shipments Same Day or Next

NEC
OKIDATA
APPLE
TELEVIDEO
COMMODORE
RADIO SHACK
AMDEK
ATARI

DIABLO 630	$2099
INTERTEC SUPERBRAIN 64K RAM	$2799
QD SUPERBRAIN	$3195
NEC 5510 SPINWRITER	$2495
NEC 5530 SPINWRITER	$2495
OKIDATA MICROLINE-80	$ 399
OKIDATA MICROLINE-82	$ 529
OKIDATA MICROLINE-83	$ 769
APPLE II PLUS 48K	$1189
APPLE DISK w/3.3 DOS Controller	$ 545
APPLE DISK w/o Controller	$ 435
BASE II PRINTER	$ 599
HAZELTINE 1420	$ 799
NORTHSTAR HORIZON II 32K QD	$2975
ANADEX DP-9500	$1295
TELEVIDEO 912C	$ 669
TELEVIDEO 920C	$ 729
TELEVIDEO 950	$ 959
CBM 8032 COMPUTER	$1225
CBM 8050 DISK DRIVE	$1449
CBM 4032 COMPUTER	$1090
CBM 4040 DISK DRIVE	$1090
CBM 4022	$ 679
CBM VIC-20	$ 289
CBM C2N	$ 85
RADIO SHACK II 64K	$3245
RADIO SHACK III 16K	$ 839
LEEDEX/AMDEK 100	$ 139
LEEDEX/AMDEK 100G	$ 169
LEEDEX/AMDEK COLOR-1 13" Color Monitor	$ 349
MICROTEK 16K RAMBOARD for Atari	$99.95
MICROTEK 32K	$ 165
ATARI 400 16K	$ 349
ATARI 825 PRINTER	$ 619
ATARI 850 INTERFACE	$ 139
or both together	$ 749
ATARI 810 DISK DRIVE	$ 449
ATARI 800 32K	$ 749
Call for price list of ATARI software	
NEC 12" MONITOR	$ 229

PRICES ARE SUBJECT TO CHANGE WITHOUT NOTICE.

WEST COAST
1-800-235-3581

OMEGA SALES CO.
3533 Old Conejo Rd. #102
Newbury Park, CA 91320
1-805-499-3678
CA. TOLL FREE 1-800-322-1873

ΩMEGA SALES CO.

EAST COAST
1-800-556-7586

OMEGA SALES CO.
12 Meeting St.
Cumberland, RI 02864
1-401-722-1027

✓89

ΩMEGA SALES COMPANY

representations in Tables 1 and 2. The savings in time and effort using this one-step method is appreciable when writing a lengthy program in machine hex code.

Tables 1 and 2 provide the key to relating hex object code to assembly mnemonics. These two tables must be used in conjunction with Tables 4–9. Each byte of information is represented in machine code by two hex digits. Thus 09, A5 and FF represent three distinct bytes of code. The exact value of each of the two digits representing a byte of code may always be the same for some instructions (EXX, for example, always has the code D9). In other cases, the value of each digit will depend on the register involved (r), the register pair involved (rr), the condition of one of the flag registers (cc) or which single bit (b) out of the eight bits in a specified register or memory location you wish to consider.

In the generalized machine codes listed in Tables 4–9, *nn* represents a two-hex-digit byte, while *nn nn* represents two successive bytes, usually the address of a memory location, with the low order byte listed first.

To illustrate the use of Tables 1 and 2 in conjunction with the remaining tables, consider the second instruction in Table 4, LD r,n. The *r* represents a single register. Suppose we wish the instruction to read LD E,06H, or "load a hex 6 into register E." The machine code for this instruction is given as (r1)(r2+6) nn. From Table 1 we find that r1 = 1 and r2 = 8 for register E. So the hex code for LD E,06 is (1)(8+6) 06, or 1E 06.

A more complicated example is one in which two registers in hexadecimal are involved, LD r,r', for which the generalized hex code is (r1+4)(r2+r'3), where r'3 is r3 for the second named register in the assembly-language instruction. For LD C,E we find from Table 1 that for register C, r1 = 0 and r2 = 8, while for register E, r3 = 3. Thus, the hex code for LD C,E is (0+4)(8+3), or 4B.

LD	r,n	(r1)(r2+6) nn
LD	(HL),0FBH	36 FB
LD	(IX+4),0FBH	DD 36 04 FB
LD	(IY+0AH),0ACH	FD 36 0A AC

(Note: In Radio Shack's Editor Assembler, a hex number beginning with a letter (A–F) must be preceded by a zero. Also, hex numbers, where different from their decimal counterparts, must be followed with an H.)

Example 1.

LD	r,r'	(r1+4)(r2+r'3)
LD	(HL),L	75
LD	(IX+15H),L	DD 75 15
LD	(IY+15H),L	FD 75 15

Example 2.

Bit Set, Reset and Test:

SET	b,r	CB (b1+12)(b2+r3)	[xxxxxx] Specified bit is set.
RES	b,r	CB (b1+8)(b2+r3)	[xxxxxx] Specified bit is reset.
BIT	b,r	CB (b1+4)(b2+r3)	[U41UOx] The complement of the specified bit is placed in Z flag.

Bit Rotate and Shift Group:

RLCA	07	[xx0x07]	RLC	r	CB 0(r3)	[220307]
RLA	17	[xx0x07]	RL	r	CB 1(r3)	[220307]
RRCA	0F	[xx0x08]	RRC	r	CB 0(r3+8)	[220308]
RRA	1F	[xx0x08]	RR	r	CB 1(r3+8)	[220308]
SLA	r	CB 2(r3)	[220307]			
SRL	r	CB 3(r3+8)	[220308]			
SRA	r	CB 2(r3+8)	[220308]			
RLD		ED 6F	[33070x]			
RRD		ED 67	[33070x]			

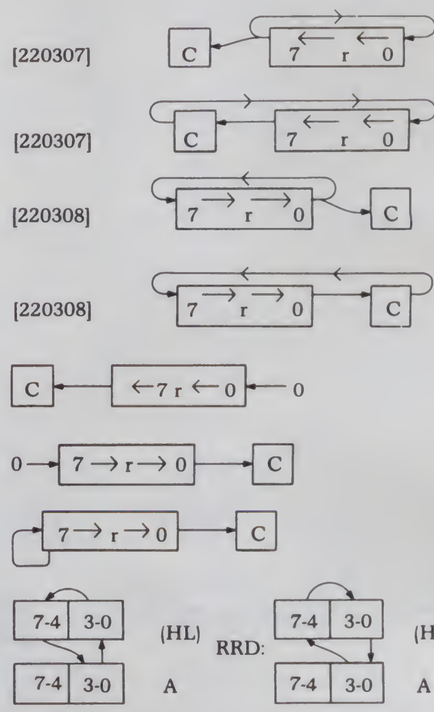


Table 6. Z-80 instructions involving bit manipulations. In these instructions, r may be (HL), (IX+d) or (IY+d), with the displacement byte dd *immediately following CB in the hex code. In the box diagrams for the rotate and shift group,* r *refers to the register,* C *refers to the single bit value of the carry flag, and the numbers refer to the actual bits in the register under consideration.*

The index registers IX and IY are not listed in Tables 1 and 2. They have not been forgotten, however. Either IX or IY can substitute for register pair HL in many instructions. Specific exceptions are presented in the notes following each table. When IX is to be used in an instruction, the hex code for HL is used, preceded by the byte *DD*. When IY is to be used, the hex code for HL is used, preceded by the byte *FD*. Table 4 shows that the 16-bit load code for LD HL,(nn) is 2A nn nn. The code for LD IX,(nn) is DD 2A nn nn; for LD IY,(nn) it is FD 2A nn nn. Note that in these cases, four bytes are required for the index registers while only three were required for the HL instruction.

Indirect addressing allows manipulations with memory contents without first loading them into a register in the CPU. This type of addressing is indicated in assembly-language mnemonics by enclosing the register pair containing the memory address in parentheses. (HL) is read as "the contents of the memory location whose address is currently contained in the HL register pair." The index registers also employ an indirect addressing mode, but with an additional displacement vector. Thus (IX+d) or (IY+d) can be used in most places that use (HL). The hex code for the index registers will again use the hex

DAA	27	[varies]	Instruction required for BCD arithmetic.
CPL	2F	[xx1x1x]	A is inverted (one's complement).
NEG	ED 44	[223615]	A is negated (two's complement).
CCF	3F	[xxxx1 –]	Carry flag is inverted.
SCF	37	[xx0x01]	Carry flag is set.
NOP	00	[xxxxxx]	No operation.
HALT	76	[xxxxxx]	CPU halts until subsequent reset or interrupt is received.

Table 7. Z-80 instructions dealing with CPU control.

code specified in Table 1 for (HL), preceded by the byte *DD* or *FD*, and followed by a one byte value *dd* representing the displacement. Examples 1 and 2 illustrate how this is done.

In cases such as these, where the index register is used with a displacement vector, the code will contain two more bytes than the corresponding code for the (HL) instruction.

Condition codes are used for the jump, call and return operations of the jump group. For these the program counter is altered only if the given condition is met; that is, only if the specified flag in the F register has the desired value of 0 (reset) or 1 (set). An example of the use of Table 1 with a conditional jump instruction is JP NZ,43FAH, which tells the program counter to change from its current value of 43FA if the Z flag (bit 6 of the F register) is not set, or 0. The hex code for this instruction from Table 4 is (c1 + 12)(c2 + 2) nn nn. Table 1 lists c1 = 0 and c2 = 0 for the NZ, not zero, condition. So the hex code for JP NZ,43FAH is C2 FA 43. (Don't forget to write the low order byte in the address first!)

DI	F3	Disables maskable interrupt.	EI	FB	Enables maskable interrupt.
IM 0	ED 46	Set interrupt mode zero.	RETN	ED 45	Return from non-maskable interrupt.
IM 1	ED 56	Set interrupt mode one.	RETI	ED 4D	Return from interrupt.
IM 2	ED 5E	Set interrupt mode two.	RST p	(p1 + 12)(p2 + 7)	Jump to one of 8 "page zero" memory locations.

Location:	00H	08H	10H	18H	20H	28H	30H	38H
p1,p2:	0,0	0,8	1,0	1,8	2,0	2,8	3,0	3,8

Table 8. Z-80 interrupt operations. No flag changes occur during processing of these instructions. The eight preselected memory locations and corresponding p1 and p2 values are shown above.

Of the six flags in the F register, the only ones ever tested for conditional operations are the zero flag (zero/not zero), the carry flag (carry/not carry), the parity/overflow flag (parity even/parity odd) and the sign flag (positive/negative). Table 3 indicates which of the two conditions for each flag results in a set condition. The JR instructions are limited to the zero and carry flags for conditional jumps.

Bit manipulations are given in Table 6. Within a given register or memory location is stored one byte of information. This byte is made of eight bits of machine code in binary (zeros and ones), and the position of a specific bit is given by a number from 0 to 7. To set bit 6 in register A, the assembly-language code is SET 6,A. Table 6 gives the corresponding hex code as CB (b1 + 12)(b2 + r3). Referring to Table 1, bit 6 has b1 = 3, b2 = 0 and register A has r3 = 7. So SET 6,A is CB (3 + 12)(0 + 7), or CB F7.

✓See List of Advertisers on page 210

A note of caution must be made here regarding the index registers used with all instructions in Table 6. The code for SRL (HL) is CB 3(r3 + 8), or CB 3E. The code for SRL (IY + d) is FD CB dd 3E. The displacement vector byte comes immediately after the CB rather than after the 3E, which is the case for instructions using the index registers in the other tables.

The rotate and shift instructions in Table 6 are best described graphically using the box diagrams shown in the table itself. In all cases, the carry flag (C in the diagrams) is involved. Rotate left (RL) and shift left (SL) operations push the value of bit 7 into the carry flag bit, while rotate right (RR) and shift right (SR) operations store the value of bit 0 into the carry flag bit. RLD and RRD shift four-bit "nibbles" around between the lower half of the A register and the two halves of the memory location pointed to by the address in the HL register pair, (HL).

Conclusion

The set of tables resulting from this effort can be fit on two pages and are more comprehensive than I had originally intended. After using them for a short time, it becomes a simple matter to merely jot down hex codes alongside assembly-language mnemonics as the program is being written, with only an occasional reference to the assembly-language manual for more detailed information regarding an instruction. I have my set of tables in two clear plastic covers facing each other so I can just open them out on the desk for reference whenever I am writing code.■

I would like to express thanks to Dr. Marvin DeJong, author of Programming & Interfacing the 6502 *(Howard W. Sams & Co., Inc., 1980), for his editorial suggestions while I was preparing this manuscript. I would also like to acknowledge use of the Radio Shack* Editor/Assembler User Instruction Manual *and program and the* Mostek Z-80 Micro-Reference Manual *from which I obtained most of my material regarding the Z-80 instruction set.*

Notes to Tables 1–10

1. Two byte addresses, nn nn, are listed with the low order byte first. Thus address 47FA would be listed in hex code as FA 47.

2. Relative jump instructions, JR, and index addressing both use displacement vectors *d*, a signed two's complement giving the displacement from the next op code address in the program. For a JR backwards, count backwards from FF (in hex), starting with the byte specifying *d*.

IN	r,(C)	ED (r1 + 4)(r2)	[22030x]				
OUT	(C),r	ED (r1 + 4)(r2 + 1)	[xxxxxx]				
	B to top 1/2 of address bus, C to bottom 1/2. One byte is transferred from device at port C to r (IN) or reverse (OUT).						
IN	A,(n)	DB nn	[xxxxxx]	OUT	(n),A	D3 nn	[xxxxxx]
	A to top 1/2 of address bus, n to bottom 1/2. One byte is transferred from device at port n to A (IN) or reverse (OUT).						
INI		ED A2	[U5UU1x]	OUTI		ED A3	[U5UU1x]
INIR		ED B2	[U1UU1x]	OTIR		ED B3	[U1UU1x]
	B to top 1/2 of address bus, C to bottom 1/2. One byte is transferred from device at port C to (HL) (IN) or reverse (OUT). Then HL is incremented and B is decremented. For INIR and OTIR, the process is repeated until B = 0.						
IND		ED AA	[U5UU1x]	OUTD		ED AB	[U5UU1x]
INDR		ED BA	[U1UU1x]	OTDR		ED BB	[U1UU1x]
	As INI, OUTI, etc., except that the value of HL is decremented.						

Table 9. Z-80 I/O instructions. The descriptions of operations are generalized and should not be taken as the exact order in which the indicated operations take place. The reader should refer to the manual for exact details. In this set, r cannot be (HL), (IX + s) or (IY + d).

	Normal Operations:				Indirect Addressing Operations:				
					[with (rr)]		[with (nn)]		[with (IX + d) or (IY + d)]
	1 byte	2 byte	3 byte	4 byte	1 byte	2 byte	3 byte	4 byte	
8-Bit Load Group, 8-Bit Arithmetic Group, Logic Group, and CP	1,4	2,7			2,7	3,10	4,13		5,19
8-Bit INC & DEC, CPU Instructions (Table 7), Interrupts, Rotate and Shift Group, SET & RESet	1,4	2,8			3,11	4,15			6,23
RLD, RRD		5,18							
BIT		2,8				3,12			5,20
16-Bit Load Group, 16-Bit INC & DEC	1,6	2,10	3,10	4,14			5,16	6,20	
Jump Group (JP & JR)		3,12	3,10		1,4	2,8			
Exchange Group	1,4				5,19	6,23			
Block Transfer, Search, and I/O (no repeat)		4,16				3,11			
Block Transfer, Search, and I/O (with repeat)		5,21							
CALL			5,17						
16-Bit Arithmetic Group, PUSH and RST	3,11	4,15							
POP and RET	3,10	4,14							

Table 10. Machine cycles and total clock periods for each instruction type. Machine cycles is given first, then total clock periods. For a 4 MHz clock, each clock period is roughly 0.25 μs; for the TRS-70 1.774 MHz clock, each clock period is 0.5638 μs.

THE NEW LAREDO LS525.® WHEN YOU'RE READY TO GET DOWN TO BUSINESS.

Laredo Systems, Inc. introduces the LS525 five megabyte rigid disk memory system for TRS80 Models I and III. Laredo's LS525 rounds up a Seagate ST506 Drive, LDOS Operating System by Logical Systems, and Laredo's own LSI

500 Series Controller into a single-board controller that emulates the famed IBM 3370 disk system, complete with:

- Improved read/write/seek access time through full block buffering and variably tuned interleave.
- On-board data separator, micro-code diagnostics and write-precompensation.
- Increased disk media reliability and data integrity with automatic defect block allocation and extended ID fields.
- Versatile CPU compatibility provided by separate host adaptor.

Introductory Offer To TRS80 Users.

Now through October 1, 1981 buy the LS525 for the OEM/dealer price of $2990, a savings of $760 off the usual suggested retail price of $3750.

LDOS Operating System with manual and diskette available for $149.

For more information about the LS525 Memory System and the LSI 500 Series Controller, contact:

Laredo Systems, Inc.
669 Giraudo Drive, San Jose, CA 95111
(408) 629-2283

laredo systems inc.

✓394

✓See List of Advertisers on page 210

Get a line on this high-quality unit available assembled or in kit form.

Plotting with the Heath X-Y Recorder

By Bob G. Roberts

Do you need an X-Y plotter for some of those BASIC application programs you've been working on? But does the price tag—or the thought of interfacing to your 6800/09 system—scare you? Then perhaps the Heath model IR-5207 X-Y recorder is what you've been looking for.

First, how much does it cost? The Heath recorder is priced at about \$480 (kit) or \$795 (assembled). Take note of the price differential; they're trying to tell you something. This is definitely not the choice I'd advise for a first-time kit builder, or for the mechanically inept. If you're at all in doubt, buy the assembled version, or enlist the aid of an experienced technician.

But what's the reward for your time spent slaving over a hot soldering iron? It's an instrument of excellent specifications, hard to match at two to three times the kit price.

The recorder is necessarily a heavy instrument when compared to others. The writing table and the entire enclosure is of one-eighth inch steel. Add the weight of the transformer and other components, and you have about 30 pounds.

However, this rugged construction provides the dimensional stability and rigidity in the writing table and pen carriage which allows the machine to meet its specifications. These specifications let you move the pen full-scale in under one second with an accuracy of ±0.5 percent.

The recorder consists of a pen carriage which moves across the 8 1/2 by 11 inch writing surface under the control of the X and Y servos. The X and Y input voltages to the recorder are filtered (up to 60 dB), amplified (maximum sensitivity is approximately 1 mV/inch), and then applied to each servo. An internally generated ramp voltage can be switched to either input, to produce a time-based record.

Five convenient sweep speeds are available, from 0.02 to 2 inches/second. At the end of each push-button-initiated sweep, the pen is lifted and the carriage is zero-positioned. A manual-reset permits a sweep to be aborted prematurely. This is particularly handy when using the slow sweep speeds. When paper is placed on the writing table, a vacuum hold-down motor keeps it in place. Thus, the machine may be used in either the horizontal or vertical position. The pen may be lifted from or

Fig. 1. Interfacing a single analog channel to the X-Y recorder.

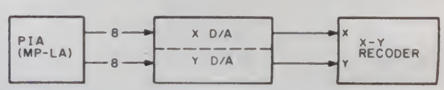


Fig. 2. Two output channels permit true X-Y plotting.

```
10 REM PIA INITIALIZATION
20 REM ROUTINE FOR PIAS
30 REM LOCATED AT $801C
40 REM AND $801E (DECIMAL
50 REM 32796 AND 32798).
60 REM
70 LET P=32796
80 POKE(P,255):POKE(P+2,255)
90 POKE(P+1,4):POKE(P+3,4)
100 END
```

Listing 1. This routine configures the PIA (MP-LA assumed) for two eight-bit output ports. Results will be the same if commands are entered in direct mode.

The Heath X-Y recorder.

Address correspondence to Bob G. Roberts, 11860 E. Fair Oak, Baton Rouge, LA 70815.

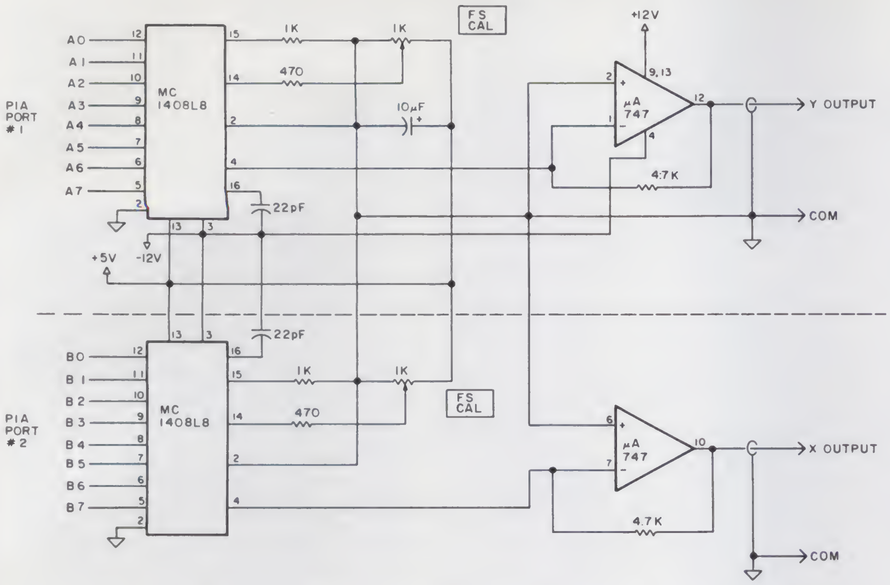


Fig. 3. Schematic diagram of two eight-bit conversion channels.

dropped to the paper with a front-panel switch.

Now for the good part. On the rear apron of the instrument (or the top, if it's vertical) are the X and Y analog input terminals. But in addition, there's a 12-pin ribbon connector which permits remote control of a number of recorder functions. TTL-level inputs here can, for example, raise and lower the pen, initiate a sweep, or mute (disable) either servo. Outputs from the connector include a null voltage from each servo, and a sample of the internal ramp voltage. There'll be more about the remote connector later.

Construction

Now, let's take a look at the construction of the recorder. Along with my instruction manual came several pages of revisions. The construction proceeds with the fabrication of the filter, preamp and main circuit boards.

Those of you who have built Heathkits before know that their assembly instructions are unexcelled in clarity. This recorder kit from Heath is no exception, and the electrical assembly, even of the main chassis, should entail no problems, although it is time-consuming.

However, the mechanical assembly, including the mounting of guide rails for the pen carriage, the pen carriage itself, the servo motors, drive cables and cable pulleys is something else. These elements must be mechanically aligned with respect to each other, cable tensions adjusted and so on. When you finally complete the electrical calibration, you'll be glad for the rugged steel frame, which, hopefully, will preserve all your mechanical adjustments. Nevertheless, you'll be much impressed with the value inherent in your completed recorder and convinced that it was all worth the effort.

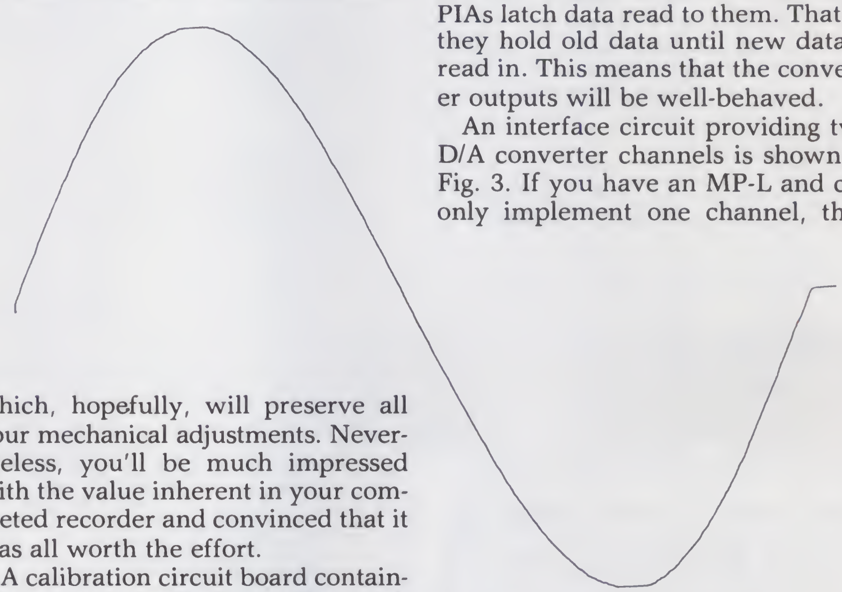
A calibration circuit board containing an accurate absolute voltage reference is mounted in the recorder, and is used to establish the basic accuracy of the front panel range switches. The recorder can be calibrated in terms of volts/inch or volts/cm before final assembly. The X or Y gain may be continuously varied between ranges, by selecting an uncalibrated mode. This mode is probably the one you will choose in your plotter applications.

To interface to the X-Y recorder, you will require at least one channel of digital-to-analog (D/A) conversion, as shown in Fig. 1. This figure shows the eight bits available from a single PIA (peripheral-interface adapter) connected to a D/A converter, which outputs an equivalent analog voltage to the recorder.

If this analog output is applied to one axis while the recorder is swept along the remaining axis, a plot of the analog voltage versus time will be obtained. But a more flexible arrangement is as shown in Fig. 2. Here, two ports of a PIA are employed with two D/A converters to obtain, logically enough, two independent analog outputs. Now, you can obtain a true X-Y plot of the analog equivalent voltage of one port with respect to that of the other.

The SWTP 6800/09 PIA may be either version MP-L or MP-LA. If yours is an MP-LA, it can easily be programmed to provide two eight-bit output channels by jumper selection on the board (see the instructions set for the board). If the PIA is an MP-L, one channel is output and one is input and the two-channel approach of Fig. 2 is denied to you (unless you have two MP-Ls, of course). Both PIAs latch data read to them. That is, they hold old data until new data is read in. This means that the converter outputs will be well-behaved.

An interface circuit providing two D/A converter channels is shown in Fig. 3. If you have an MP-L and can only implement one channel, then



Sine wave output of Listing 2.

```
100 REM SINE PLOTTER ROUTINE
110 REM * DO INITIALIZATION FIRST! *
120 REM
130 P1=3.14159265:D=P1/180
140 REM PLOTTING LOOP
150 FOR I=0 TO 360
160 LET A=I*D
170 F=125*SIN(A)
180 F=127+SGN(F)*INT(ABS(F))
190 POKE(P,F):REM 'P' AS IN LISTING 1.
200 NEXT I
210 END
```

Listing 2. Only one D/A converter is required to plot a sine curve on the recorder, if the remaining axis is generated with the internal sweep.

that portion below the dashed line can be deleted. The arrangement employs the MC1408L8 eight-bit D/A converter integrated circuit, which currently costs about $6. The uA747 I/C, a dual uA741, converts the current output of the 1408L8 to a voltage ranging from zero to almost 10 V in 1/256-volt increments. For more details on the 1408L8, see the manufacturer's application note.

Power requirements for the two-channel version are less than 50 mA for the 5 V supply, and less than 30 mA for the ±12 V supplies. Construction layout is not critical, but the 22 pF compensation capacitor should be connected as close to pins 16 and 3 of the 1408L8 as possible. When construction is complete, adjust the "FS CAL" potentiometers to yield an output of 10 V from each channel, when the digital inputs are all 1s (5 V). If the circuit is operating properly, all 0s as digital inputs will produce 0 V output.

If all has gone well, interconnect the computer PIAs, D/A channels and X-Y plotter as previously shown. Be sure to consult the Heath manual for instructions on strapping the common input terminals to the recorder. Verify all connections before applying power to anything.

Before data can be sent to the plotter, the PIA must be configured properly. This is done in BASIC as shown in Listing 1. Each PIA port used must be configured prior to use, and anytime the computer reset is operated. Next, use the POKE (addr,data) command of BASIC to zero both converter channels. For example, if your converter is at hex location $801C:

POKE(32796,0)

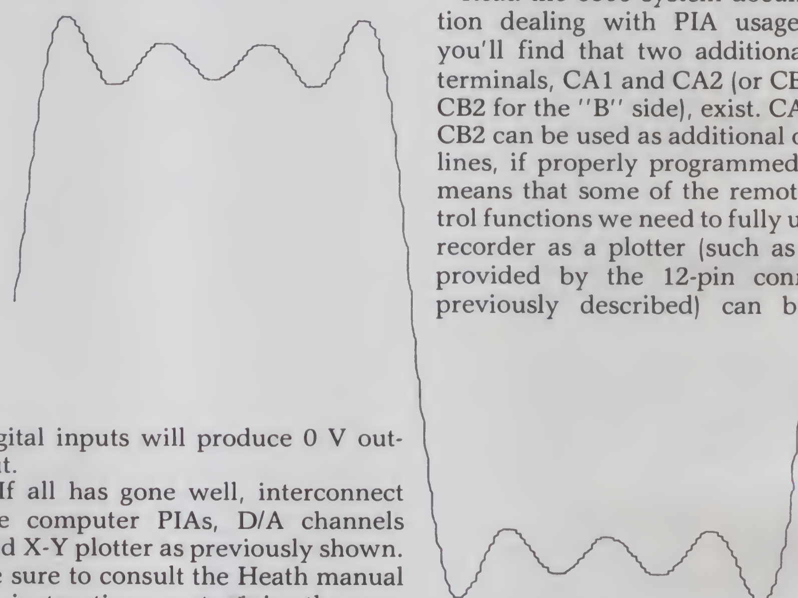
should produce zero volts from the converter connected to that location. Accordingly, if the recorder is in the Servo mode, the pen holder should move to the zero position. With the recorder range switches in the 1000 mV positions, send a full-scale voltage to it:

POKE(32796,255)

The recorder servo should overrange. Use the CAL-VAR switch and the VAR screw-adjust to set the pen holder to the on-scale position you wish to be full-scale. Try obtaining intermediate deflections, using the POKE command, to gain familiarity and confidence. Error reports will be issued by BASIC if you try to send values that are negative or that exceed 255 to the converter.

So, herein lies the rub: successful use of the converter requires that the entire range of X and Y values dispatched to it lie between zero and 255. If you must have better resolution than 1/256 full-scale, this is not the converter for you. Much can be done, however, within these restrictions, as you will see.

Read the 6800 system documentation dealing with PIA usage, and you'll find that two additional PIA terminals, CA1 and CA2 (or CB1 and CB2 for the "B" side), exist. CA2 and CB2 can be used as additional output lines, if properly programmed. This means that some of the remote-control functions we need to fully use the recorder as a plotter (such as those provided by the 12-pin connector previously described) can be ob-



Square-wave approximation output of Listing 3.

```
100 REM SQ. WAVE APPROXIMATION
110 REM ROUTINE FOR X-Y RECORDER.
120 REM
130 P1=3.14159265:D=P1/180
140 FOR I=0 TO 360 STEP 2
150 A=I*D
160 F1=SIN(A):F3=SIN(3*A)/3
170 F5=SIN(5*A)/5:F7=SIN(7*A)/7
180 F=F1+F3+F5+F7
190 F=127+100*SGN(F)*INT(ABS(F))
195 REM
200 REM IF Y IS AT 'P' AND X IS
210 REM AT 'P'+2 ...
220 REM
230 POKE(P,F):POKE(P+2,I/2)
240 NEXT I
250 END
```

Listing 3. This routine plots a square-wave approximation as the sum of sine terms through the 7th harmonic.

tained by appropriately using the CA2 or CB2 outputs.

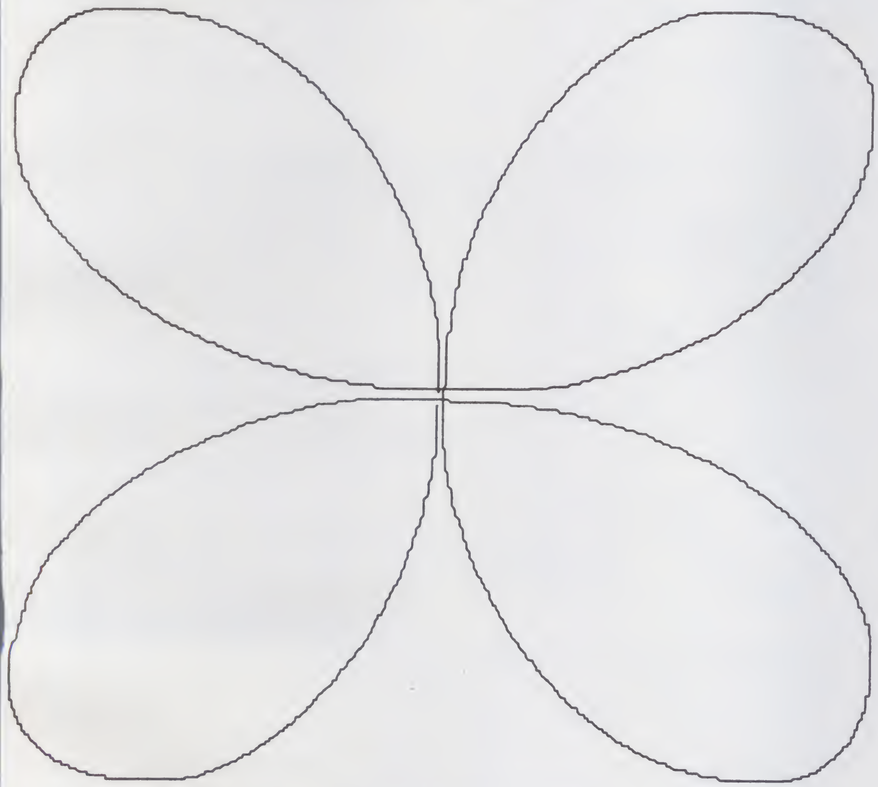
Having configured the ports and adjusted the full-scale recorder response for each channel, how can the converter be used in BASIC programs? Probably the best way to learn this is the way you learned to program originally—by examining programs and their outputs, and then doing it. Listing 2 is of a program which plots a sine curve using only the Y channel. Run is initiated with the X recorder input selector at sweep. Operate the sweep start button on the recorder, and away you go. Don't forget to position the pen and lower it first.

If you're using SWTP BASIC, the computation time for trigonometric functions is great enough to allow the pen to settle before the next value is generated. If you're using TSC BASIC, you'd better include a delay (such as an empty FOR-NEXT loop) between plots, because it's fast. You'll find the appropriate sweep range for yourself after a run or two.

Listing 3 depicts a program using both X and Y channels to plot the Fourier representation of a square wave, using harmonics through the seventh. This one is run with X and Y input selectors at 1000 mV (VAR as previously set), and the pen is lowered manually before the carriage return is typed. Note that this output could've been generated in the same manner as before, using the sweep function.

Listing 4 is used to plot a four-leaved rose, and could not have been produced conveniently using only one channel. The limitations of eight-bit data conversion are evident in the somewhat squared petals of the rose, but the results are respectable.

Now it's up to you. Analyze and run these programs yourself, and allow your imagination to take over. I'll bet that you find the Heath X-Y recorder and this converter combination to be a useful peripheral for your computer system. ■



Four-leaved rose output of Listing 4.

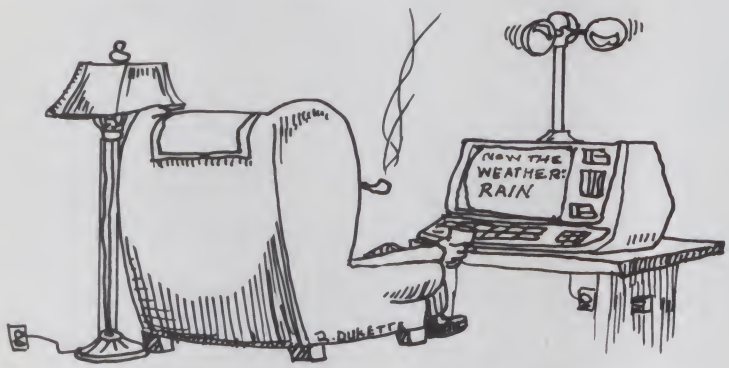
```
100 REM PROGRAM TO PLOT A
110 REM 4-LEAVED ROSE ON THE
120 REM X-Y RECORDER
130 REM
140 P1=3.14159265:D=P1/180
145 DEF FNR(Z)=SGN(Z)*INT(ABS(Z)+.5)
150 FOR I=0 TO 360
160 A=I*D
170 L=100*SIN(2*A)
180 X=L*COS(A):Y=L*SIN(A)
190 X=FNR(X):Y=FNR(Y)
200 POKE(P,Y+127):POKE(P+2,X+127)
210 NEXT I
220 END
```

Listing 4. This short program plots a four-leaved rose. The pen should be lowered manually after run is initiated.

✓See List of Advertisers on page 210

If it rains on your day off, don't say your TRS-80 didn't warn you.

Little Gem Weather Forecaster



By Mike Barlow

The weather is always a good topic of conversation, so how about using your computer to generate a forecast? You won't actually be using instruments and transducers. Instead, you'll count on recovering the subconscious information humans already know about the weather.

The program, written in TRS-80 BASIC, is self-explanatory. Just follow the prompts, and answer the questions. Oh yes, the temperature is good for the room you are in at present, and nobody knows what a kilopascal is anyway. Have fun and amaze your friends! ■

Address correspondence to Mike Barlow, 5052 Chestnut Ave., Pierrefonds, Quebec H8Z 2A8, Canada.

Program listing.

```
10 'LITTLE GEM WEATHER FORECASTER BY MIKE BARLOW JULY 1980
20 CLS
30 PRINT@80,"LITTLE GEM WEATHER FORECASTER"
40 PRINT@149,"* * * * * * * * * *"
50 GOSUB200:GOSUB180:CLS
60 PRINT@12,"PLEASE ANSWER THE FOLLOWING QUESTIONS:"
70 GOSUB200:PRINT@ 143,"IS THE SUN SHINING (Y/N)?":GOSUB190:IFA$="Y"THENS1$=
"SUNNY"ELSES1$="CLOUDY"
80 PRINT@271,"DOES IT LOOK LIKE STAYING THAT WAY (y/n)?":GOSUB190:IFA$="Y"TH
EN S2$="ALL DAY"ELSE S2$=" PERIODS"
90 PRINT@399,"IS THERE A WIND (Y/N)?":GOSUB190:IFA$="Y"THENW1$="WINDS"ELSEW1
$="LIGHT WINDS"
100 PRINT@527,"IS IT GETTING STRONGER (Y/N)?":GOSUB190:IFA$="Y"THENW2$=" INC
REASING"ELSE W2$=" SLOWLY DECREASING"
110 PRINT@655,"DOES IT LOOK LIKE RAINING (Y/N)?":GOSUB190:IFA$="Y"THENR1$="C
HANCE OF RAIN"ELSER1$="LIKELIHOOD OF RAIN IS LESS THAN TEN PER CENT"
120 PRINT@783,"CAN YOU HEAR THUNDER (Y/N)?":GOSUB190:IFA$="Y"THENT1$=" THUND
ERSTORMS AND HIGH WINDS APPROACHING"ELSET1$="CHANCE OF THUNDERSTORMS LATER"
130 CLS:PRINT@81,"LItTLE GEM WEATHER FORECAST"
140 PRINT@144,STRING$(29,"-"):GOSUB200
150 PRINT@260,"THE OUTLOOK FOR THE REST OF TODAY IS ";S1$;" ";S2$;", ";W1$;W
2$;", ";R1$;"; ";T1$;"."
160 GOSUB200:PRINT"    THE BAROMETRIC PRESSURE IS 41.8 KILOPASCALS AND RISIN
G. THE TEMPERATURE HERE IS 21 DEGREES CELSIUS, OR 74 DEGREES FAHRENHEIT"
170 GOTO170
180 PRINT@977,"PRESS <ENTER> TO CONTINUE";
190 A$=INKEY$:IFA$=""THEN190ELSERETURN
200 FORI=1TO999:NEXT:RETURN
```

'Universatility.' It's as easy as 1. 2. 3.

'Universatility' can be found in three versions of Synertek Systems' SYM single board computer — the versatile, universal evaluation board.

Over 20,000 SYM-1 boards have been used for learning about and evaluating 6502 microprocessors for specific applications. OEM SYM boards are used in hundreds of products.

Now Synertek Systems presents the new SYM-1/68 for 6802 microprocessors, and the SYM-1/69 for 6809 microprocessors. These boards are designed to reliably perform *the same functions* as the SYM-1 board for these popular microprocessors. Each SYM board is complete and ready-to-use with its own version of the 4K byte ROM SUPERMON monitor firmware.

Modification kits are also available to quickly and easily convert existing SYM-1 boards to SYM-1/68 or SYM-1/69 microcomputers.

Build on your microprocessor knowledge with the 'universatility' of SYM microcomputer boards from Synertek Systems.

Every SYM-1, SYM-1/68, and SYM-1/69 single board computer features:

- 28 double-function keypad with audio response
- 4K byte ROM resident SUPERMON with over 30 standard monitor functions and expandable for individual requirements
- Up to 24K bytes of available program memory
- Expansion to 4K bytes of on-board static RAM
- Expansion to 71 Input/Output lines
- Single +5V power supply required
- Standard interfaces for audio cassette with remote control, 185 bytes/second cassette format, TTY and RS-232-C, system expansion bus, four I/O buffers, and oscilloscope single-line display

Synertek Systems ✓375

P.O. Box 552 Santa Clara, CA 95052
Telephone: (408) 988-5689
TWX: 910-338-0135

Dealer inquiries invited.

Find out the answer to this question fast with this CBASIC-2 calendar program that keeps you up to date.

What's On for Tomorrow?

By Ray Vukcevich

Here's a straightforward way to use your computer as an appointment and things-to-do calendar. CALEN calculates leap years, displays or prints the calendar as weeks, and lets you add, delete, change and switch data by date. The program is written in CBASIC-2, and program notes for converting and modifying the program for your own special needs are included.

The program treats each year as a separate random access file with fixed fields. When the program is run, the year is first requested and verified:

Year (ie 1981)? 1980
Is 1980 correct? [RETURN]

Then the main menu is displayed:

Calendar
N—Set Up New Year
Y—Change Year
S—See
C—Change Data
Task?

First you will need to set up a year file. After entering the year, select N from the main menu. The year will be displayed highlighted, and you will be asked for the day number of Jan. 1:

JAN 1 Day Number (Monday is 1)?

For example, January 1 of 1980 is a Tuesday, or day number 2. Enter 2 for JAN 1 Day Number for 1980, and enter 4 for 1981. (1981 began on a Thursday.) The program will then set up the year file. It takes several minutes. You can't write over a year file by mistake. An error message, YEAR "already exists," will be displayed if you try to create a year file over an old one. You must first delete the old file at the operating system level. If you change your mind and decide not to create a new year file, simply hit RETURN, and the program will return to the main menu.

The Y function is for changing years at the main menu level. If you have two year files, you will use Y to move from one to the other. Or if you enter the wrong year, you can go back with the Y function.

The See choice on the main menu lets you examine and print the calendar by week. Enter a starting date, and then step through as much of the calendar as you wish. If you reach the end, the program will start again with Jan. 1.

When the month changes while displaying the week, the new month will be displayed at the top of the screen and a highlighted asterisk (*) will appear next to the first day of the new month. (See Sample 1.)

Use the C function to add, delete, change or switch data. You are first

```
Oct 1980
Tue 30                                Wed 1 *
------------------------------------  ------------------------------------
------------------------------------  ------------------------------------
------------------------------------  ------------------------------------
```

Sample 1.

```
Jan 1980
Tue 1
---------------------------------------------------------------------------
---------------------------------------------------------------------------
---------------------------------------------------------------------------
(B-Blank W-Write S-Switch N-New Date RETURN-Next Q-Quit)
Task?
```

Sample 2.

Address correspondence to Ray Vukcevich, 1036 E. Orange #20, Tempe, AZ 85281.

asked for a date:

Date (MM,DD)?

Then the data already in that date is displayed. If it is empty, it looks like Sample 2.

You can write three lines using W, you can delete any line or all of them with B and you can switch the data in this entry with data in another date with S. Hitting RETURN in this case displays data for Jan. 2; N asks for another date, and Q takes you back to the main menu. RETURN, N and Q all write the data displayed on the screen before moving on.

The Program

Line numbers are necessary in CBASIC only as labels for branching destinations—that is, the places where GOTO and GOSUB statements send control. See lines 43 and 46 for examples. The line numbers on the extreme left in the listing were compiler generated and were not typed in when the program was written.

CHR$(126) + CHR$(28) in line 3 clears the screen and homes the cursor on a Hazeltine 1500. CHR$(126) + CHR$(31) causes the Hazeltine to display highlighted characters, and CHR$(126) + CHR$(25) returns the display to normal. Use your own terminal's codes.

Long variables (FORM.FEED$, for example) are OK in CBASIC. Also notice the use of \ in line 122. The back slash indicates to the CBASIC compiler that the line is to be continued.

Look at IF ERSW1% THEN in line 49. This means "if ERSW1% is not equal to zero," or in this case, "if ERSW1% is equal to 1."

You will need to rewrite the read and write statements (lines 222 and 227) to conform with random access forms in your BASIC.

Modifications

If you want more lines in each date entry, change line 10 to accommodate the larger record size. Then simply add a LINE.4$ after each occurrence of LINE.3$ and adjust the display format.

Another change you may want to make is to have the program continue with another year after reaching December 31. You will need to open another file beginning perhaps at line 117.

It is true that you can buy database management programs to handle your calendar, but they are expensive. CALEN does the job well, is quick and is easily implemented.■

```
Nov 1980

Mon 27                                  Tue 28
1. Call Fenster -- 2. Bills             1. Bentor 11 am
3. Write Larry                          2. Loan Company 2:30 pm
4. Finish Russ Project                  3. Ms Martin 4:30 pm

Wed 29                                  Thr 30
   NEW CONTACTS                         1. Mail card to Mary
      FIRST                             2. Call Bob
     DISTRICT                           ------------------------------

Fri 31                                  Sat 1 ~*~
1. Mail final forms to                  1. Rent  2. Loan payment
Sam and Martin                          3. Car
2. Call Denny back                      ------------------------------

Sun 2
REST!
READ.
RELAX!
```

Sample appointment calendar.

Program listing. Calendar program in CBASIC-2.

```
 1: REM APPOINTMENT CALENDAR
 2: REM BY RAY VUKCEVICH
 3:              CLEAR$=CHR$(126)+CHR$(28)
 4:              BRIGHT$=CHR$(126)+CHR$(31)
 5:              DIM$=CHR$(126)+CHR$(25)
 6:              FORM.FEED$=CHR$(12)
 7:              CODE$="NSCQ"
 8:              BLANK$="                              "
 9:              DASH$="------------------------------"
10:              REC.SIZE%=110
11:              DIM DAY$(7),MONTH$(12),FIRST%(12),DAYS%(12)
12:              DIM LINE.1$(8),LINE.2$(8),LINE.3$(8),WK.DY$(8)
13:         FIRST%(1)=1:FIRST%(2)=32:FIRST%(3)=60:FIRST%(4)=91
14:         FIRST%(5)=121:FIRST%(6)=152:FIRST%(7)=182:FIRST%(8)=213
15:         FIRST%(9)=244:FIRST%(10)=274:FIRST%(11)=305:FIRST%(12)=335
16:         DAYS%(1)=31:DAYS%(2)=28:DAYS%(3)=31:DAYS%(4)=30
17:         DAYS%(5)=31:DAYS%(6)=30:DAYS%(7)=31:DAYS%(8)=31
18:         DAYS%(9)=30:DAYS%(10)=31:DAYS%(11)=30:DAYS%(12)=31
19:         DAY$(1)="Mon":DAY$(2)="Tue":DAY$(3)="Wed":DAY$(4)="Thr"
20:         DAY$(5)="Fri":DAY$(6)="Sat":DAY$(7)="Sun"
21:         MONTH$(1)="Jan":MONTH$(2)="Feb":MONTH$(3)="Mar"
22:         MONTH$(4)="Apr":MONTH$(5)="May":MONTH$(6)="Jun"
23:         MONTH$(7)="Jul":MONTH$(8)="Aug":MONTH$(9)="Sep"
24:         MONTH$(10)="Oct":MONTH$(11)="Nov":MONTH$(12)="Dec"
25: REM GET YEAR
26: 5       PRINT CLEAR$
27:         PRINT:PRINT:PRINT:PRINT
28:         INPUT "Year (ie 1981)?";YEAR$
29:         PRINT "Is ";BRIGHT$;YEAR$;DIM$;" correct";
30:         INPUT LINE Q$
31:         IF LEFT$(Q$,1)="N" THEN 5
32: REM CHECK FOR LEAP YEAR
33:         YEAR=VAL(YEAR$)
34:         A=YEAR/4
35:         B=INT(A)
36:         C=A-B
37:         IF C<>0 THEN 10
38:         DAYS%(2)=29
39:         FOR X%=3 TO 12
40:              FIRST%(X%)=FIRST%(X%)+1
41:         NEXT X%
42: REM GET MENU SELECTION
43: 10      GOSUB 20
44:         ON T% GOSUB 100,200,300,400
45:         GOTO 10
46: 20      PRINT CLEAR$
47:         PRINT
48: REM ERROR MESSAGES
49:         IF ERSW1% THEN ERSW1%=0:PRINT TAB(10);BRIGHT$;\
50:              "Year not found";DIM$:PRINT
51:         IF ERSW2% THEN ERSW2%=0:PRINT TAB(10);BRIGHT$;\
52:             YEAR$;" already exists";DIM$:PRINT
53:         PRINT TAB(10);"Calendar"
54:         PRINT TAB(10);"--------"
55:         PRINT
56:         PRINT TAB(10);"N - Set Up New Year"
57:         PRINT TAB(10);"Y - Change Year"
58:         PRINT
59:         PRINT TAB(10);"S - See"
60:         PRINT TAB(10);"C - Change Data"
61:         PRINT
62:         PRINT TAB(10);"Q - Quit"
63:         PRINT
64:         INPUT "Task?";T$
65:         T$=LEFT$(T$,1)
66:         IF T$="Y" THEN 5
67:         FOR T%=1 TO 5
```

More

Listing continued.

```
                IF T$=MID$(CODE$,T%,1) THEN RETURN
        NEXT T%
                GOTO 10
REM SET UP NEW YEAR
100     PRINT CLEAR$
        PRINT:PRINT BRIGHT$;YEAR$;DIM$:PRINT
        PRINT "JAN 1 Day Number (Monday is 1)";
        INPUT LINE DAY.NUM$
        IF DAY.NUM$="" THEN RETURN
        DAY.NUM%=VAL(DAY.NUM$)
        IF DAY.NUM%<1 OR DAY.NUM%>7 THEN 100
        PRINT:PRINT BRIGHT$;"Writing...";DIM$
        READ%=0
                GOSUB 1000
                IF OLD.FILE% THEN ERSW2%=1:GOTO 110
                LINE.1$=DASH$:LINE.2$=DASH$:LINE.3$=DASH$
                FOR X%=1 TO FIRST%(12)+DAYS%(12)
                        REC.NUM%=X%
                        DAY$=DAY$(DAY.NUM%)
                        GOSUB 1200
                        DAY.NUM%=DAY.NUM%+1
                        IF DAY.NUM%>7 THEN DAY.NUM%=1
                NEXT X%
110     CLOSE 1
                                                RETURN
REM SEE
200     PRINT CLEAR$
        PRINT:PRINT
        INPUT "Starting Date (MM,DD)?";NUM%,DATE%
        IF NUM%<1 OR NUM%>12 THEN 200
        IF DATE%<1 OR DATE%>DAYS%(NUM%) THEN 200
        START%=(FIRST%(NUM%)-1)+DATE%
        READ%=1:NEW%=0
        GOSUB 1000
        IF OLD.FILE%=0 THEN ERSW1%=1:RETURN
205             FOR X%=1 TO 7
                        REC.NUM%=START%
                        GOSUB 1100
                        DATE$=STR$(DATE%)
                        IF NEW%=0 THEN WK.DY$(X%)=DAY$+" "+DATE$
                        IF NEW% THEN NEW%=0:WK.DY$(X%)=DAY$+" "+\
                                  DATE$+" "+BRIGHT$+"*"+DIM$
                        LINE.1$(X%)=LINE.1$
                        LINE.2$(X%)=LINE.2$
                        LINE.3$(X%)=LINE.3$
                        DATE%=DATE%+1
                        START%=START%+1
                        IF DATE%>DAYS%(NUM%) THEN DATE%=1:NUM%=NUM%+1:\
                                                NEW%=1
                        IF START%>FIRST%(12)+30 THEN START%=1:\
                                      NUM%=1
                NEXT X%
REM DISPLAY WEEK
        PRINT CLEAR$
210     IF TASK$="P" THEN PRINT:PRINT:PRINT:PRINT TAB(5);MONTH$(NUM%);\
                                  " ";YEAR$:PRINT:GOTO 215
        PRINT TAB(5);BRIGHT$;MONTH$(NUM%);" ";YEAR$;DIM$:PRINT
215     FOR X%=1 TO 7 STEP 2
                PRINT TAB(5);WK.DY$(X%);TAB(45);WK.DY$(X%+1)
                PRINT TAB(5);LINE.1$(X%);TAB(45);LINE.1$(X%+1)
                PRINT TAB(5);LINE.2$(X%);TAB(45);LINE.2$(X%+1)
                PRINT TAB(5);LINE.3$(X%);TAB(45);LINE.3$(X%+1)
                PRINT
        NEXT X%
        IF TASK$="P" THEN PRINT FORM.FEED$:CONSOLE
        PRINT TAB(5);"(P-Print  RETURN-Next  Q-Quit)"
        PRINT TAB(5);"Task";
        INPUT LINE TASK$
        TASK$=LEFT$(TASK$,1)
        IF TASK$="Q" THEN 220
        IF TASK$="P" THEN LPRINTER:GOTO 210
                GOTO 205
220                                             CLOSE 1
                                                READ%=0
                                                RETURN
REM CHANGE DATA
300     READ%=1
        GOSUB 1000
        IF OLD.FILE%=0 THEN ERSW1%=1:RETURN
310     PRINT CLEAR$
        INPUT "Date (MM,DD)?";NUM%,DATE%
        IF NUM%<1 OR NUM%>12 THEN 310
        IF DATE%<1 OR DATE%>DAYS%(NUM%) THEN 310
320     REC.NUM%=(FIRST%(NUM%)-1)+DATE%
                GOSUB 1100
330     PRINT CLEAR$
        PRINT:PRINT:PRINT
        PRINT TAB(20);BRIGHT$;MONTH$(NUM%);" ";YEAR$;DIM$
        PRINT
        PRINT TAB(20);DAY$;DATE%
        PRINT TAB(20);LINE.1$
        PRINT TAB(20);LINE.2$
        PRINT TAB(20);LINE.3$
        PRINT
        PRINT TAB(5);"(B-Blank  W-Write  S-Switch  N-New Date  ";
        PRINT "RETURN-Next  Q-Quit)"
        PRINT TAB(5);"Task";
```

More

✓See List of Advertisers on page 210

Listing continued.

```
          INPUT LINE TASK$
          TASK$=LEFT$(TASK$,1)
          IF TASK$="N" OR TASK$="" THEN 350
          IF TASK$="Q" THEN 360
          IF TASK$="B" THEN GOSUB 1300
          IF TASK$="W" THEN GOSUB 1400
          IF TASK$="S" THEN 340
                GOTO 330
340       INPUT "Switch Date (MM,DD)";SM%,SD%
          IF SM%<1 OR SM%>12 THEN 340
          IF SD%<1 OR SD%>DAYS%(SM%) THEN 340
          REC.1%=REC.NUM%
          DAY.1$=DAY$
          HOLD.1$=LINE.1$
          HOLD.2$=LINE.2$
          HOLD.3$=LINE.3$
          REC.NUM%=(FIRST%(SM%)-1)+SD%
                GOSUB 1100
                DAY.2$=DAY$
                REC.2%=REC.NUM%
                DAY$=DAY.1$
                REC.NUM%=REC.1%
                GOSUB 1200
          DAY$=DAY.2$
          LINE.1$=HOLD.1$
          LINE.2$=HOLD.2$
          LINE.3$=HOLD.3$
                REC.NUM%=REC.2%
                GOSUB 1200
                REC.NUM%=REC.1%
                GOSUB 1100
                GOTO 330
350       GOSUB 1200
          IF TASK$="N" THEN 310
          DATE%=DATE%+1
          IF DATE%>DAYS%(NUM%) THEN DATE%=1:NUM%=NUM%+1
          IF NUM%>12 THEN NUM%=1
          GOTO 320
360       GOSUB 1200
                                                      CLOSE 1
                                                      READ%=0
                                                      RETURN
REM QUIT
400       PRINT CLEAR$
                                                      STOP
REM OPEN FILE
1000      OLD.FILE%=0
          IF END #1 THEN 1010
          OPEN YEAR$ RECL REC.SIZE% AS 1
          OLD.FILE%=1
          GOTO 1020
1010      IF READ% THEN 1020
          CREATE YEAR$ RECL REC.SIZE% AS 1
1020                                                  RETURN
REM READ RECORD
1100      END.FILE%=0
          IF END # 1 THEN 1110
          READ # 1,REC.NUM%;DAY$,LINE.1$,LINE.2$,LINE.3$
                                                      RETURN
1110      END.FILE%=1
                                                      RETURN
REM WRITE RECORD
1200      PRINT # 1,REC.NUM%;DAY$,LINE.1$,LINE.2$,LINE.3$
                                                      RETURN
REM BLANK LINES
1300      PRINT TAB(5);"Line (1, 2, 3 or ALL)";
          INPUT LINE LN.NUM$
          IF LN.NUM$="" THEN 1310
          LN.NUM$=LEFT$(LN.NUM$,1)
          IF LN.NUM$="A" THEN LINE.1$=DASH$:\
                LINE.2$=DASH$:LINE.3$=DASH$
          IF LN.NUM$="1" THEN LINE.1$=DASH$
          IF LN.NUM$="2" THEN LINE.2$=DASH$
          IF LN.NUM$="3" THEN LINE.3$=DASH$
1310                                                  RETURN
REM WRITE LINES
1400      PRINT "Line #1";"------------------------------*"
          INPUT "       ";LINE LN$
          IF LN$="" THEN 1410
          LINE.1$=LEFT$(LN$+BLANK$,30)
1410      PRINT "Line #2";"------------------------------*"
          INPUT "       ";LINE LN$
          IF LN$="" THEN 1420
          LINE.2$=LEFT$(LN$+BLANK$,30)
1420      PRINT "Line #3";"------------------------------*"
          INPUT "       ";LINE LN$
          IF LN$="" THEN 1430
          LINE.3$=LEFT$(LN$+BLANK$,30)
1430                                                  RETURN
END
NO ERRORS DETECTED
CONSTANT AREA:       8
CODE SIZE:        3277
DATA STMT AREA:      0
VARIABLE AREA:     432
```

80 COLUMN GRAPHICS

The Integrated Visible Memory for the PET has now been redesigned for the new 12" screen 80 column and forthcoming 40 column PET computers from Commodore. Like earlier MTU units, the new K-1008-43 package mounts inside the PET case for total protection. To make the power and flexibility of the 320 by 200 bit mapped pixel graphics display easily accessible, we have designed the Keyword Graphic Program. This adds 45 graphics commands to Commodore BASIC. If you have been waiting for easy to use, high resolution graphics for your PET, isn't it time you called MTU?

K-1008-43M Manual only $10 (credited toward purchase)
k-1008-43 Complete ready to install package $495

Mastercharge and Visa accepted

Write or call today for our full line catalog describing all MTU 6502 products, including our high speed 8" Floppy Disk Controller for up to 4 megabytes of PET storage.

The image on the screen was created bv the program below.

```
 10 VISMEM: CLEAR
 20 P=160: Q=100
 30 XP=144: XR=1.5*3.1415927
 40 YP=56: YR=1: ZP=64
 50 XF=XR/XP: YF=YP/YR: ZF=XR/ZP
 60 FOR ZI=-Q TO Q-1
 70 IF ZI<-ZP OR ZI>ZP GOTO 150
 80 ZT=ZI*XP/ZP: ZZ=ZI
 90 XL=INT(.5+SQR(XP*XP-ZT*ZT))
100 FOR XI=-XL TO XL
110 XT=SQR(XI*XI+ZT*ZT)*XF: XX=XI
120 YY=(SIN(XT)+.4*SIN(3*XT))*YF
130 GOSUB 170
140 NEXT XI
150 NEXT ZI
160 STOP
170 X1=XX+ZZ+P
180 Y1=YY-ZZ+Q
190 GMODE 1: MOVE X1,Y1: WRPIX
200 IF Y1=0 GOTO 220
210 GMODE 2: LINE X1,Y1-1,X1,0
220 RETURN
```

MTU

154

Micro Technology Unlimited
2806 Hillsborough Street
P.O. Box 12106
Raleigh, NC 27605, U.S.A.
(919) 833-1458

NOW 80 COLUMN PETS CAN HAVE MTU HIGH RESOLUTION GRAPHICS

If you're an overburdened clerical worker, get your boss to implement this Heath program.

Relief for the Hassled Clerk

By Kathleen and D. C. Shoemaker

Many workers see automation as a threat to their job security. It is therefore important that an employer who is installing micros create a receptive atmosphere.

One way is to find an application for a task that everyone sees as pure drudgery. For instance, many low-ranked clerks spend endless hours keeping track of time cards, time spent on jobs for cost accounting, and so on. If you can use a micro to ease their burdens, you can win them over, and break the barrier that sometimes scuttles the most well-intentioned efforts.

This program, called Timekeeping, is just such an application. It keeps track of the time spent by a given worker on a particular job in a particular shop.

The information is recorded by the shop foreman on a time card, one for each worker, and turned in to the accounting office at the end of each work day. For example, in the company we were working with, each employee had a clock number that he or she used to punch in and out. Each job was assigned a unique number for cost accounting purposes, and there were four shops—electrical, mechanical, rebuild and machine.

For output, the chief accountant wanted a recap of each worker's total hours for each two-week pay period, the number of hours worked on each job and the number of hours worked in each shop. This information became the basis of the payroll submission, and was used to cost out each job, both in direct and indirect labor hours.

The program needed to be user-

```
10 REM *** Erase H19 screen
20 PRINTCHR$(27);"E"
30 :
40 PRINT"                         Timekeeping Program":PRINT
50 PRINT"    This set of programs allows you to enter timekeeping information"
60 PRINT"by Clock and Job Number, hours worked and the department worked."
70 PRINT"There are limited error-trapping facilities, but you should be careful"

80 PRINT"to enter the data correctly.  You always have the option of pressing"
90 PRINT"CTRL-C and restarting the program by typing RUN.":PRINT
100 :
110 REM *** This routine allows the user to alter the date in HDOS memory
120 REM *** without returning to HDOS.  Useful for bookkeeping programs.
130 ON ERROR GOTO510
140 :
150 PRINT:PRINT"    The date currently in the computer is ";:GOSUB430
160 LINEINPUT"    Enter the desired date (DD-MMM-YY) <no change>:  ";D$
170 IFLEN(D$)<2THEN220
180 GOSUB400
190 :
200 REM  Menu
210 :
220 PRINT:PRINT"    You may choose from the following:":PRINT
230 PRINT"              (1)  Enter and sort daily data"
240 PRINT"              (2)  Sort and join files for summarization"
250 PRINT"              (3)  Summarize files by Job or Clock Number"
260 PRINT"              (4)  Formatted print routine"
270 PRINT"              (5)  Exit from the Introduction to BASIC"
280 PRINT"              (6)  Exit from the Introduction to HDOS"
290 PRINT:INPUT"    What is your choice";C
300 IF C=1THENLOAD"ENTER",R
310 IF C=2THENLOAD"WEEKSORT",R
320 IF C=3THENLOAD"SUM1",R
330 IF C=4THENLOAD"PRINT",R
340 IF C=5THENPRINT:PRINT"    Program ends.  Returning to BASIC.":PRINT:END
350 IF C=6THENPRINT:PRINT"    Exiting to HDOS.":PRINT:SYSTEM
360 PRINT:PRINT"    Entry error.  Try again.":GOTO0220
370 :
380 REM *** Routine for poking the new date into memory
390 :
400 FOR J=8383TO8391
410 D3$=MID$(D$,J-8382,1)
420 POKE J,ASC(D3$)
430 NEXT J
440 PRINT"    The date is now set as ";
450 :
460 REM *** Retrieve date from HDOS and print
470 :
480 D2$="":FOR L=8383TO8391:D2$=D2$+CHR$(PEEK(L)):NEXT L
490 PRINT D2$
500 RETURN
510 PRINT"    You've typed something wrong.  Try again.":GOTO0150
```

Listing 1. This is the menu program from which all other parts of the package are selected. It also establishes the date in the H8 memory under which the data files will be saved and retrieved.

Address correspondence to Kathleen and D. C. Shoemaker, 2000A Foxridge, Blacksburg, VA 24060.

friendly, save time and be accurate. The best approach seemed to be a menu-driven set of programs that would lead the computer operator through the sequence of steps required, and produce the desired output almost automatically.

The equipment available included a Heath H8 computer with 56K of memory, two 5-1/4-inch disk drives, an H14 printer and an H19 terminal. We wrote the package in Microsoft BASIC; formatting the output would be easier, and the programs would be more portable.

Looking at Listing 1, the menu, you can see that there are a few nonstandard, system-specific items. The date routines at lines 400 and 480 take advantage of the date stored in the H8's memory, which we used in creating data files you could subsequently identify by looking at the disk directory; you need not read the data file itself to see the date on which it was created.

This feature could be replaced or omitted if some other provision were made to assign dates to the data files.

Note, too, that the operator can either exit from the menu to BASIC or to the disk operating system (HDOS). This is to aid error recovery, and can be modified once the program is running. Sample 1 is a copy of the screen display as presented to the operator.

Listing 2 is the heart of the system; it is the program that creates and checks the data, and performs the initial sort. The operator is prompted for each entry, and at the end of each complete time card entry is asked to verify the data. Errors can be corrected before they are recorded.

Once all the data for a given date has been entered, it is saved in a set of files that will be used later by other parts of the program.

Sample 2 is a copy of what the operator sees on the screen during the running of Listing 2. There is one file for each category—clock numbers, job numbers and department names.

The quick-sort routine that places these files in sequential order is fairly efficient if the number of data entries to be sorted is under 100. This is used when the original data will later be required for some other purpose. The unsorted data can be deleted at this point, either by the operator or under program control. If disk space is no problem, however, it might be better to leave it as a backup file, since it

```
                    Timekeeping Program

     This set of programs allows you to enter timekeeping information
by Clock and Job Number, hours worked and the department worked.
There are limited error-trapping facilities, but you should be careful
to enter the data correctly.  You always have the option of pressing
CTRL-C and restarting the program by typing RUN.


     The date currently in the computer is 18-Jan-81
     Enter the desired date (DD-MMM-YY) <no change>:  21-Jan-81
     The date is now set as 21-Jan-81

     You may choose from the following:

                  (1)  Enter and sort daily data
                  (2)  Sort and join files for summarization
                  (3)  Summarize files by Job or Clock Number
                  (4)  Formatted print routine
                  (5)  Exit from the Introduction to BASIC
                  (6)  Exit from the Introduction to HDOS

     What is your choice?
```

Sample 1. This is a copy of the CRT screen displayed to the operator when the menu is called up. Note that 18 January 1981 was the date in the H8's memory at the time the menu was called, but it was changed to 21 January 1981.

Listing 2. This is the main program of the package. It allows daily time-card data to be entered, and does the initial sorting. In its present form, the original source data files are retained, but they could be automatically deleted after the sorting if disk space were a problem.

```
10 I=1:PRINTCHR$(27);CHR$(69)
20 D2$="":FOR L=8383TO8391:D2$=D2$+CHR$(PEEK(L)):NEXT L
30 D1$="":FOR L=8383TO8384:D1$=D1$+CHR$(PEEK(L)):NEXT L
40 C$="      ###":J$="              ###":H$="                ###.##       "
50 C1$="###          ":J1$="###          ":H1$="###.##          "
60 PRINT"    The data you enter will be saved under the date ";D2$;".":PRINT
70 X1=150:X2=200:X3=999:REM  X1=highest clk#,X2=max#Jobs,X3=highest Job#
80 DIM C(X2),J(X2),D$(X3),H(X2),H1(X2),H2(X3)
90 PRINT"Departments: 1=Electrical, 2=Machine, 3=Mechanical, 4=Rebuild, ";
100 PRINT"5=Stop Entry"
110 LINEINPUT"What department                                   ? ";D$(I)
120 IF D$(I)="1"THEN D$(I)="Electrical":GOTO180
130 IF D$(I)="2"THEN D$(I)="Machine   ":GOTO180
140 IF D$(I)="3"THEN D$(I)="Mechanical":GOTO180
150 IF D$(I)="4"THEN D$(I)="Rebuild   ":GOTO180
160 IF D$(I)="5"THEN270
170 PRINTCHR$(7);"    Data input error.  Re-type.":GOTO90
180 INPUT"Clock number                                  ";C(I):K=C(I)
190 IF(K<1OR K>150)THENPRINTCHR$(7);"    Data entry error. Re-type":GOTO180
200 INPUT"Job number                                    ";J(I):O=J(I)
210 IF J(I)>999THENPRINTCHR$(7);"    Data input error.  Re-type.":GOTO200
220 INPUT"How many hours on this Job                    ";H(I)
230 PRINT:LINEINPUT"    Is this entry CORRECT (y or n) <y>?";A$
240 IFLEFT$(A$,1)="n"THENPRINT"    Re-type the data.":GOTO90
250 H1(K)=H1(K)+H(I):H2(O)=H2(O)+H(I)
260 I=I+1:PRINT:GOTO90
270 F$="SY1:CLK"+D1$+".DAT"
280 OPEN"O",#1,F$
290 FOR X=1TO(I-1):PRINT#1,USING C1$;C(X);
300 PRINT#1,USING J1$;J(X);
310 PRINT#1,USING H1$;H(X);
320 PRINT#1,"          ";D$(X),D1$:NEXT
330 PRINT"    Clock data has been saved as ";F$:CLOSE#1
340 F$="SY1:JOB"+D1$+".DAT"
350 OPEN"O",#1,F$
360 FOR X=1TO(I-1):PRINT#1,USING J1$;J(X);
370 PRINT#1,USING C1$;C(X);
380 PRINT#1,USING H1$;H(X);
390 PRINT#1,"          ";D$(X),D1$:NEXT
400 PRINT"    Job data has been saved as ";F$:CLOSE#1
410 F$="SY1:DEPT"+D1$+".DAT"
420 OPEN"O",#1,F$
430 FOR X=1TO(I-1):PRINT#1,D$(X),
440 PRINT#1,USING C1$;C(X);
450 PRINT#1,USING J1$;J(X);
460 PRINT#1,USING H1$;H(X);
470 PRINT#1,"          ";D1$:NEXT
480 PRINT"    Department data has been saved as ";F$:CLOSE#1
490 CLEAR6000:N=1:DIM A$(500):PRINT:PRINT"    Sorting Routines":PRINT
500 PRINT"    Reading clock number data file..."
510 D1$="":FOR L=8383TO8384:D1$=D1$+CHR$(PEEK(L)):NEXT L
520 F$="SY1:CLK"+D1$+".DAT
530 OPEN"I",#1,F$
540 LINEINPUT#1,A$(N)
550 IF EOF(1)THEN580
560 N=N+1
570 GOTO540
580 CLOSE #1
590 PRINT"    Sorting...":PRINT
600 FOR J=1TON:FORK=1TON:IFA$(J)>A$(K)THEN620
610 SWAP A$(J),A$(K)
620 NEXT:NEXT
630 PRINT"   ";N;"clock numbers were sorted.":PRINT
640 F$="SY1:CSORT"+D1$+".DAT"
650 OPEN "O",#1,F$
660 FOR M=1TON:PRINT#1,A$(M):NEXT
670 CLOSE:PRINT"    Sorted data saved as ";F$:GOTO680
```

More

Listing 2 continued.

```
680 N=1
690 PRINT:PRINT"     Now reading job number data file..."
700 F$="SY1:JOB"+D1$+".DAT"
710 OPEN"I",#1,F$
720 LINE INPUT #1,A$(N)
730 IF EOF(1)THEN 760
740 N=N+1
750 GOTO720
760 CLOSE #1
770 PRINT"     Sorting...":PRINT
780 FOR J=1TO N:FOR K=1TO N:IF A$(J)>A$(K)THEN800
790 SWAP A$(J),A$(K)
800 NEXT:NEXT
810 PRINT"    ";N;"Job numbers were sorted.":PRINT
820 F$="SY1:JSORT"+D1$+".DAT"
830 OPEN"O",#1,F$
840 FOR M=1TON:PRINT #1,A$(M):NEXT
850 CLOSE:PRINT"     Sorted data saved as ";F$:GOTO860
860 N=1
870 PRINT:PRINT"     Now reading department data file...":
880 F$="SY1:DEPT"+D1$+".DAT"
890 OPEN"I",#1,F$
900 LINEINPUT#1,A$(N)
910 IF EOF(1)THEN 940
920 N=N+1
930 GOTO 900
940 CLOSE#1
950 PRINT"     Sorting...":PRINT
960 FOR J=1TO N:FOR K=1TO N:IF A$(J)>A$(K)THEN980
970 SWAP A$(J),A$(K)
980 NEXT:NEXT
990 PRINT"    ";N;"department entries were sorted.":PRINT
1000 F$="SY1:DSORT"+D1$+".DAT"
1010 OPEN"O",#1,F$
1020 FOR M=1TON:PRINT#1,A$(M):NEXT
1030 CLOSE:PRINT"     Sorted data has been saved as ";F$;".  Program ends."
1040 LOAD"INTRO",R
```

```
     The data you enter will be saved under the date 21-Jan-81.

Departments: 1=Electrical, 2=Machine, 3=Mechanical, 4=Rebuild, 5=Stop Entry
What department                                  ? 1
Clock number                                     ? 122
Job number                                       ? 101
How many hours on this job                       ? 11.5

     Is this entry CORRECT (y or n) <y>?y

Departments: 1=Electrical, 2=Machine, 3=Mechanical, 4=Rebuild, 5=Stop Entry
What department                                  ? 3
Clock number                                     ? 123
Job number                                       ? 119
How many hours on this job                       ? 9.25

     Is this entry CORRECT (y or n) <y>?y

Departments: 1=Electrical, 2=Machine, 3=Mechanical, 4=Rebuild, 5=Stop Entry
What department                                  ?
```

Sample 2. The CRT display while the data entry Listing 2 is in use. Note that the operator had to enter a Y at the end of the time card entry, thus offering an opportunity to correct the entry.

could always be retrieved, re-sorted and used if a sorted data file were lost.

Finally, the program returns the operator to the menu to either change the date in memory and enter the next day's data, or to go on to something else.

When the data for a pay period has been gathered, the operator can summarize each category either by clock number or job number. This is often a useful cross-check, and would be tedious if not for the next set of programs.

Listing 3 selects the desired files and concatenates them for summarization. The product is a temporary file called "SORTDATE.DAT," where the date is the current date in the H8's memory. Since this is a temporary file, you don't really need to worry about the date, but under certain circumstances it might be a good idea to keep the file for backup.

After the concatenated files have been created, the summarization programs can be run. These are in two parts because of conflicting variables; it was easier to create another temporary file than to worry about accidental collisions of data.

Listing 4, the first part, reads each presorted entry, and, if the current one is the same as the last one, sums the two values. If not, it begins a new account. This is then saved, and Listing 5, the second part, is called in. The temporary file is reread and printed to the device specified. You are then returned to the menu.

The last program, Listing 6, is a simple formatting routine that per-

mits printing of a sorted clock-number file. This particularly interested the chief accountant of the company we were working with.

Conclusion

It would be relatively easy for the owner of a small business to enter these programs into his or her personal computer for that particular application. But if you're interested in the concepts behind this package, we recommend that you look around your own office and see what similar operations are being done manually—jobs requiring a lot of data manipulation and processing, or formatting.

While your application may not be just like this one, chances are there are some similar, repetitive clerical functions that could be done better, faster and with greater accuracy. Even if no time can be saved, the increase in accuracy and the more usable format of the finished product may make it worthwhile.

Once a small business commits itself to using microcomputers, many other applications will suggest themselves, as soon as the people involved become used to the idea of a computer in the office. ■

```
10 PRINTCHR$(27);"E":CLEAR(16000):DEFINT A-Z:DIM A$(350):I=1
20 D1$="":FOR L=8383TO8384:D1$=D1$+CHR$(PEEK(L)):NEXT L
30 ON ERROR GOTO310
40 PRINT"    This program allows you to identify a series of data files to be"
50 PRINT"sorted, as for a week's worth of entries.  Type the name in the form"
60 PRINT"'nnnxx' where 'nnn' represents the type of file (e.g., clk) and 'xx'"
70 PRINT"represents the date the file was created (e.g., 17).  Do not type the"
80 PRINT"system number (SY1:) or the extension (.DAT) as these will be added"
90 PRINT"automatically.":PRINT
100 LINEINPUT"    Where do you want to put the sorted file <SY1:>?  ";W$
110 LINEINPUT"    Enter the name of the file to sort:  ";M$
120 IF W$=""THEN W$="SY1:SORT"+D1$+".DAT"
130 M1$="SY1:"+M$+".DAT":OPEN"I",1,M1$
140 IF EOF(1)THEN160
150 LINEINPUT#1,A$(I):PRINT I;:I=I+1:GOTO140
160 CLOSE:PRINT:LINEINPUT"    Do you have another file <n>?  ";B$
170 IFLEFT$(B$,1)="y"THEN110
180 C=I:B=I
190 C=INT(C/2):PRINT C;:IF C=0THEN270
200 D=1:E=B-C
210 F=D
220 G=F+C:IF A$(F)<=A$(G)THEN250
230 SWAP A$(F),A$(G):F=F-C:IF F<1THEN250
240 GOTO220
250 D=D+1:IF D>E THEN190
260 GOTO210
270 OPEN"O",#1,W$
280 FOR X=2TO I
290 PRINT#1,A$(X):NEXT X:CLOSE
300 LOAD"INTRO",R
310 PRINT"    Error in file name or file non-existent.  Re-enter.":GOTO110
```

Listing 3. This program combines daily sorted data files for use with the summarization programs, or for storage as weekly or biweekly records.

```
10 PRINTCHR$(27);"E";"    List Summarization Program":PRINT
20 PRINT"    This program prints a summarization of Clock or Job Number files."
30 PRINT"depending on the file name you enter.  To run, enter the name of the"
40 PRINT"data file you want.  Use the following format:":PRINT
50 PRINTTAB(25);"FILENAME":PRINT
60 PRINT"where 'FILENAME' may be any file name you want, such as 'SORT01'"
70 PRINT"or 'CSORT22'.":PRINT
80 CLEAR16000:N=1:DIM A$(400)
90 Q=1000:REM  Set Q for range of Job Numbers
100 DIM K(Q),S(Q)
110 LINEINPUT"    Enter the name of the file to summarize:  ";F1$
120 F$="SY1:"+F1$+".DAT"
130 PRINT:PRINT"    Reading ";F$;" file...  ";
140 N=1
150 OPEN "I",#1,F$
160 LINEINPUT#1,A$(N)
170 IF EOF(1)THEN200
180 N=N+1
190 GOTO160
200 CLOSE#1:PRINT"done.":PRINT
210 REM  Read the values to be compared and summarized
220 FOR M=1TO N
230 K(M)=VAL(LEFT$(A$(M),3)):REM  Get a value for the first Job Number
240 S(M)=VAL(MID$(A$(M),23,6)):REM  Get a value for the first Hours Worked
250 NEXT M
260 PRINT"    Writing the temporary data file...  ";
270 OPEN"O",#1,"SY1:TEMP.DAT"
280 FOR M=1TON
290 PRINT#1,K(M),S(M)
300 NEXT M
310 CLOSE#1:PRINT"done."
320 LOAD"SUM2",R
```

Listing 4. This is the first of a two-part program to read a file created by Listing 3, and summarizes either the job numbers or the clock numbers.

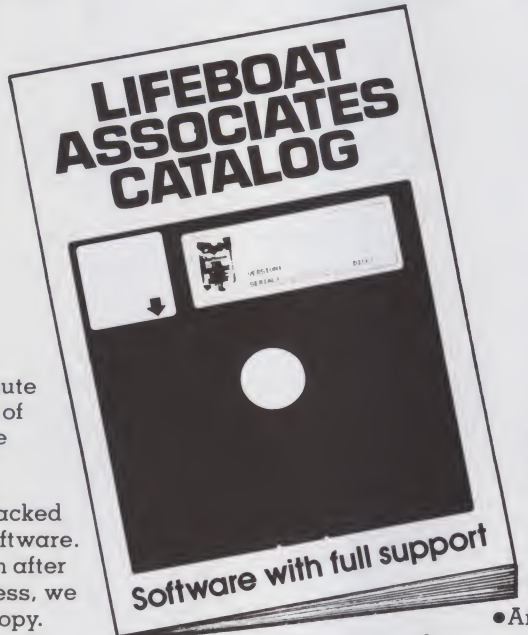
```
10 REM  This is the second part of the summary program
20 CLEAR6000:Q=2000:PRINT
30 DIM K(Q),S(Q)
40 LINEINPUT"    Where do you want the summary (TT:,LP: or a disk):  ";D$
50 PRINT:LINEINPUT"    Name the print-out:  ";N$
60 PRINT"    Now reading temporary data file for summing...":PRINT
70 X$="###":Y$="###.##"
80 OPEN"I",#1,"SY1:TEMP.DAT"
90 N=1000
100 INPUT#1,K(N),S(N)
110 IF EOF(1)THEN140
120 N=N+1
130 GOTO100
140 CLOSE#1
150 OPEN"O",#1,D$
160 PRINT#1,TAB(20);N$:PRINT#1,:PRINT#1,:
170 PRINT#1,TAB(12);"Number                     Hours"
180 PRINT#1,TAB(12);"-------                    -----":PRINT#1,:
190 REM  Subtotal routine
200 M=1000
210 S(K(M))=0
220 S(K(M))=S(K(M))+S(M)
230 M=M+1:IF M>(N+1)THEN330
240 IF K(M)=K(M-1)THEN220
250 IF K(M)<>K(M-1)THEN GOSUB280
260 GOTO210
270 REM  Printing subroutine
280 PRINT#1,TAB(15);
290 PRINT#1,USING X$;K(M-1);
300 PRINT#1,"          ";
310 PRINT#1,USING Y$;S(K(M-1))
320 RETURN
330 CLOSE:LOAD"INTRO",R
```

Listing 5. The second part of the summarization program. This is a separate program to prevent the data collision possible if the entire summarization is done at one time.

```
10 PRINTCHR$(27);"E";TAB(20);"Formatted File Print Routine":PRINT
20 PRINT"    This program will print a sorted Clock Number file to the line"
30 PRINT"printer, causing each clock number to be printed on a separate page."
40 CLEAR16000:M=1:DIM A$(350):PRINT
50 LINEINPUT"    Enter the name of the file to print (NAMEDD):  ";D1$
60 D$="SY1:"+D1$+".DAT"
70 OPEN"I",#1,D$
80 LINEINPUT#1,A$(M)
90 IF EOF(1)THEN110
100 M=M+1:GOTO80
110 CLOSE#1
120 M=M+1:N=1
130 P$="LP:"
140 OPEN"O",#1,P$
150 PRINT#1,A$(N)
160 N=N+1
170 IF LEFT$(A$(N),3)<>LEFT$(A$(N-1),3)THENPRINT#1,CHR$(12)
180 IF N=MTHEN190ELSE GOTO150
190 CLOSE#1
200 LOAD"INTRO",R
```

Listing 6. This prints a formatted dump of the clock number file on the printer (or any other device named in line 160). It keeps each clock number on a separate page to facilitate record checking according to the office procedures in the company for which it was originally written. It could have been done in a variety of ways.

When it comes to microcomputer software we wrote the book.



How do you stay up-to-the-minute with the rapidly changing world of microcomputer software? Get the Lifeboat Catalog.

The Latest Innovations

The new Lifeboat Catalog is packed with the latest state-of-the-art software. And if we publish a new program after the latest catalog has gone to press, we enclose a flash bulletin in your copy.

The Greatest Selection

Because Lifeboat is the world's largest publisher of microcomputer software, our catalog offers you the greatest selection of programs for business, professional and personal use. Our more than 200 programs range from integrated accounting and professional practice systems to office tools for bookkeepers and secretaries to sophisticated tools for programmers. Included are business systems, word processors, programming languages, database management systems, application tools and advanced systems utilities.

We specialize in software that runs on most small business computers. Our more than 60 media formats, including floppy disks, data cartridges, magnetic tape and disk cartridges, support well over 100 computer brands of domestic and foreign manufacture.

Get Full Support

We give the crucial dimension of after-sales service and full support to everything we sell. That includes:

- A telephone hotline where technical and customer service representatives will answer your questions 11 hours a day.
- An update service for software and documentation.
- Telephone, telex and mail-order service at the New York headquarters and at overseas offices in England, France, Switzerland, West Germany and Japan.
- Export service providing software delivery to six continents.
- Subscriptions to *Lifelines*™, the monthly magazine that offers comparative reviews, tips, techniques, identified bugs and updates that keep you abreast of change.

Get It Now

Lifeboat now serves tens of thousands of satisfied customers with our breadth of up-to-date, fully tested, fully supported and competitively priced software.

You may not need all we offer, but we offer just what you need. After all, we wrote the book.

LIFEBOAT WORLDWIDE offers you the world's largest library of software. Contact your nearest dealer or Lifeboat:

Lifeboat Associates
1651 Third Ave.
New York, N.Y. 10028
Tel: (212) 860-0300
Telex 640693 (LBSOFT NYK)
TWX 710-581-2524

Lifeboat Inc.
OK Bldg., 5F
1-2-8, Shiba-Daimon
Minato-ku, Tokyo 105, Japan
Tel: 03-437-3901
Telex: 2423296 (LBJTYO)

Lifeboat Associates, Ltd.
PO Box 125
London WC2H 9LU, England
Tel: 01-836-9028
Telex: 893709 (LBSOFTG)

Lifeboat Associates GmbH
PO Box 168, Aegeristrasse 35
CH 6340 Baar, Switzerland
Tel: 042-31-2931
Telex. 865265 (MICO CH)

Intersoft GmbH
Schlossgartenweg 5
D-8045 Ismaning, W. Germany
Tel: 089-966-444
Telex 5213643 (ISOFD)

Lifeboat Associates, SARL
10, Grande Rue Charles de Gaulle
92600 Asnieres, France
Tel: 1-733-08-04
Telex: 250303 (PUBLIC X PARIS)

✓312

Lifeboat Associates
Software with full support

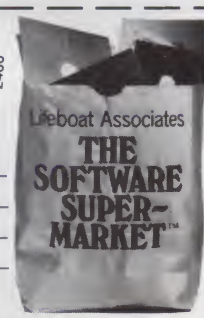
Mail coupon to: Lifeboat Associates, 1651 Third Avenue, New York, New York 10028 or call (212) 860-0300.

2406

☐ Please send me a free Lifeboat Catalog.
☐ Please send me *Lifelines* subscription information.

Name ____________ Title ____________
Company ____________
Street ____________
City ____________ State ____________ Zip ____________

Lifelines is a trademark of Lifelines Publishing Corporation.



Let Your Micro Speak-2-U-2

By Chris Wieland

Like many computer hobbyists, I've always wanted to add a speech synthesizer to my system, but haven't had the money. Happily, this situation has recently been remedied. The Percom Data Company has a circuit board called Speak-2-Me-2 that interfaces Texas Instruments' Speak & Spell to the TRS-80.

Now, that may be exciting to many of you, but I'm sure the rest are saying: "Swell, another interesting device for all those other people who have TRS-80s. But what about my Apple (or PET, or OSI, or home-brew system)?" Well, don't give up yet. As long as you have an eight-bit parallel port (or, more specifically, five bits of output and one of input) and can do a little machine-language programming, you too can use the Speak-2-Me-2.

Specifically, Percom's interface card gives you access to the nearly 300-word vocabulary of the Speak & Spell learning aid. This word list includes the numbers 0 through 10, the 26 letters of the alphabet, and a variety of nouns and verbs which can be used in making up your own sentences. The words are spoken in the same clear voice heard in normal operation of the Speak & Spell, but now your computer controls the order in which they are spoken.

How the Speak & Spell Does It

Believe it or not, there are only four integrated circuits (ICs) on the Speak & Spell circuit board, and two of these are identical ROM chips. Of the remaining ICs, one is a modified version of TI's popular four-bit microprocessor chip, the TMS 1000, and the other is TI's speech chip, or the TMS 5100.

The speech synthesizer chip and the two ROM chips form the heart (or rather voice) of the Speak & Spell machine. The microprocessor chip, or controller, is merely used to tell the synthesizer chip which word to speak next. (It also monitors the keyboard and controls the alphanumeric display seen in normal use of the Speak & Spell.)

It is the speech synthesizer chip that does all the work associated with speech synthesis. Fortunately, TI designed its speech synthesizer chip to be interfaced with most microprocessors, and thus it is fairly simple to replace the Speak & Spell's controller with whatever processor chip your system is currently running on.

Taking Control of Your Speak & Spell

What occurs during synthesis of a word is that a 16-bit address is sent by the controller to the synthesizer chip four bits at a time. The synthesizer chip uses this address to look up a string of data from the ROMs which it uses for the synthesis of speech.

The synthesizer chip uses a form of speech synthesis known as linear predictive coding (LPC). This coding method allows for a relatively slow data rate of 2400 bits per second (i.e., the rate new information must be obtained from the ROM storage during speech synthesis) while still maintaining a high quality of speech. (See Refs. 1 and 2 for a detailed description of the synthesizer chip's internal architecture and speech coding method.)

Address correspondence to Dr. Chris Wieland, Dept. EPO Biology, University of Colorado—Boulder, Boulder, CO 80309.

Listing 1. 6502 machine-language driver for controlling Speak & Spell talking machine via Percom's Speak-2-Me-2 interface. Driver is written for a microprocessor with a 1 MHz clock. Parts labeled as "STUFF" are just that; that is, steps necessary for the Speak & Spell to run, but their function is not explained in Percom's instruction manual.

```
*****MACHINE LANGUAGE DRIVER FOR SPEAK-2-ME-2 INTERFACE*****
*****USING A 6502 PROCESSOR AND TWO 8-BIT PARALLEL PORTS****

     0000                START   ORG $0F00
     0000                STORE   EQU $0F60        ***STORAGE FOR POKED #S
     0000                OUTPOR  EQU $F100        ***8 BIT OUTPUT PORT
     0000                INPORT  EQU $F102        ***8 BIT INPUT  PORT

***********************************************************
***********************************************************

     0F00    A2 05       RESET   LDX #$05         **********************
     0F02    A9 00               LDA #$00         ***  SEND OUT FIVE  ***
     0F04    20 40 0F    AGAIN   JSR STROBE       *** ZEROS TO CLEAR  ***
     0F07    CA                  DEX              ***  SPEAK & SPELL  ***
     0F08    D0 F8               BNE AGAIN        **********************
     0F0A    A9 80               LDA #$80
     0F0C    4C 40 0F            JMP STROBE       *** RETURN TO BASIC ***
***********************************************************
     0F0F    A9 00       SPEAK   LDA #$00         ***ENTER HERE TO TALK**
     0F11    20 40 0F            JSR STROBE
```

More

Listing 1 continued.

```
0F14    A2 05               LDX #$05       ***********************
0F16    BD 60 0F   GETONE   LDA STORE,X    **SEND 16-BIT ADDRESS**
0F19    20 39 0F            JSR OUTCHR     **4 BITS AT A TIME  ***
0F1C    CA                  DEX            **THEN SEND A ZERO  ***
0F1D    D0 F7               BNE GETONE     ***********************

0F1F    A9 80               LDA #$80       ***********************
0F21    20 40 0F            JSR STROBE     ***      OTHER      ***
0F24    A9 A0               LDA #$A0       ***      STUFF      ***
0F26    20 40 0F            JSR STROBE     ***      NEEDED     ***
0F29    A9 E0               LDA #$E0       ***      TO RUN     ***
0F2B    20 40 0F            JSR STROBE     ***      SPEAK &    ***
0F2E    A9 00               LDA #$00       ***      SPELL      ***
0F30    20 40 0F            JSR STROBE     ***********************

0F33    AE 02 F1   BACKUP   LDX INPORT     ***********************
0F36    10 FB               BPL BACKUP     ***LOOP TILL NOT BUSY**
0F38    60                  RTS            ***********************

0F39    48         OUTCHR   PHA            ***********************
0F3A    A9 20               LDA #$20       ***MORE UNEXPLAINED ***
0F3C    20 40 0F            JSR STROBE     ***    STUFF        ***
0F3F    68                  PLA            ***********************

0F40    49 FF      STROBE   EOR #$FF       ***********************
0F42    49 08               EOR #$08       ***INVERT ADDRESS,SET**
0F44    8D 00 F1            STA OUTPOR     ***STROBE & WAIT 20MS**
0F47    48                  PHA            ***    DELAY        ***
0F48    68                  PLA            ***    DELAY        ***
0F49    48                  PHA            ***    DELAY        ***
0F4A    68                  PLA            ***    DELAY        ***
0F4B    EA                  NOP            ***    DELAY        ***
0F4C    49 08               EOR #$08       *** CLEAR STROBE    ***
OF4E    8D 00 F1            STA OUTPOR     ***CHANGE STROBE BIT***
0F51    60                  RTS            ***********************
```

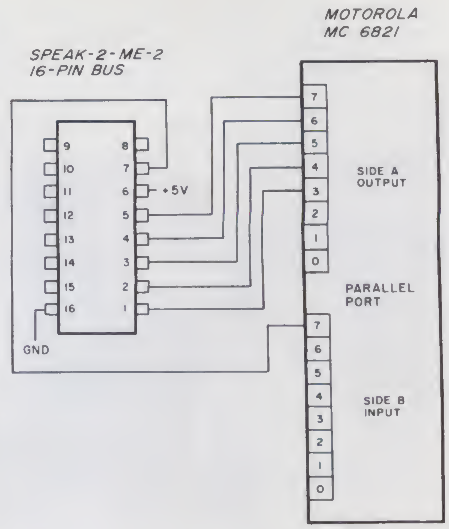


Fig. 1. Connections between the Speak-2-Me-2 interface card and a parallel port (Motorola's MC6821). Although the Speak-2-Me-2 interconnecting cable comes with a 40-pin edge connector on one end, only 16 lines are used and the other end of the cable (that plugged into the interface board) is a simple 16-pin DIP. For the above connections I cut off the 40-pin edge connector and soldered the wires directly to the parallel port.

Using the data obtained from the ROM chips, the synthesizer recreates the waveform of the previously coded word, using an on-board eight-bit digital-to-analog converter (DAC). The DAC's output signal is processed by a low pass filter, amplified, and sent to the small speaker on the front of the Speak & Spell, where it is heard as a clear male voice.

The Speak-2-Me-2 kit costs $69.95, and includes an instruction manual which shows you how to modify the Speak & Spell circuit board to disable the TI processor and replace it with your system's own microprocessor. The single-sided circuit board included with the kit performs the necessary buffering of data lines between the parallel port and the synthesizer chip, and replaces the battery-based power supply of the Speak & Spell.

Also included is the all important list of addresses which must be supplied to the synthesizer chip to say the particular word or phrase stored at that address.

For those of you with a TRS-80, Percom supplies a 40-pin interconnecting cable that hooks up to your expansion interface or to your printer cable adapter. In addition, a Level II software driver program is included in the instructions.

I should say that I have not tested the TRS-80 interface plug and software. But the interface supplied has given me no problems when used with my Ohio Scientific C1-P and home-brew Motorola 6821 parallel port.

Software and Hardware

Fig. 1 shows how I connected the Speak-2-Me-2 interconnecting cable (after cutting off the 40-pin edge connector) to my parallel port. The MC6821 by Motorola has two eight-bit parallel ports (sides A and B) in which each bit can be individually programmed to act as an input or an output. I chose to configure the full eight bits of port A as output and port B as input, but it would be possible to get by using only one of the two sides by configuring a single side with both input and output lines and slightly modifying the program logic.

Listing 1 shows the 6502 machine-language program I use as a driver program for speech synthesis. Note that the program obtains the 16-bit address of the word it should say from the four most significant bits of memory locations $0F61–0F65 (hex) which have been previously poked into place by a BASIC program (Listing 2). This BASIC program, written in OSI's Microsoft BASIC, can run with only 4K of memory, and includes lines (100–220) which poke the machine-language driver in place each time the program is run. The machine-language program is called as a user function whenever a word is to be spoken.

Note also that memory location $0F60 must contain a zero (line 50) for the program to work.

With this BASIC program you may enter up to 50 consecutive words for the Speak & Spell to say. Other important points are given as comments in Listing 2.

I have also prepared a timing diagram (Fig. 2) of the functions carried out by the machine-language program in Listing 1. This should help anyone with another microprocessor to use the Speak-2-Me-2 interface with his own parallel port. If you can program your machine to recreate the timing signals of Fig. 2, you should have no trouble in using the Speak-2-Me-2 interface to run the Speak & Spell's speech synthesizer.

Increasing Your Vocabulary

Some other possibilities which you might also consider include increasing the Speak & Spell's vocabulary with additional ROM speech modules, available from TI for under $20. Percom does not tell how to add these in their instructions, but it shouldn't be too difficult to find out what line is used to access added ROM modules during normal use of the Speak &

Spell, and then just recreate that signal with your parallel port.

Another option is the possibility of forming new words from parts of currently stored words. Percom's instruction manual tells how to stop the pronunciation of words or phrases before they are completed and suggests this as a basis for forming new words. I have tried this technique and have found it to work quite well. It can even be used to form those four-letter words needed when your current program doesn't work or when you wipe out your only copy of a disk.

(Percom also sells a minidiskette for the TRS-80 which makes use of this feature to form new words from parts of old ones. It costs $29.95 and includes eight talking game programs.)

Finally, as you may have seen in the stores where you bought your Speak & Spell, two new talking machines are available from TI: Speak & Read and Speak & Math. They each have different vocabularies than the Speak & Spell, and the Speak & Math in particular may have a more practical vocabulary for use in data entry and mathematical applications, since it can say most any number up to 999,999,999.9. However, the Percom

```
10   REM ** OSI MICROSOFT BASIC PROGRAM TO CONTROL SPEAK & SPELL
20   REM ** VIA PERCOM S SPEAK-2-ME-2 INTERFACE
30   REM ** WRITTEN BY C. WIELAND
40   DIM W(50),C(3),S(200)
50   POKE 3937,0: REM $F060=0
60   REM ** SET UP PARALLEL PORT (MC6821) AT HEX $F100 & 02
70   POKE 61697,0: POKE 61699,0
80   POKE 61696,255: POKE 61698,0
90   POKE 61697,4: POKE 61699,4
100 DATA 162,5,169,0,32,64,15,202,208,248
110 DATA 169,128,76,64,15,169,0,32,64,15
120 DATA 162,5,189,96,15,32,57,15,202,208
130 DATA 247,169,128,32,64,15,169,160,32,64
140 DATA 15,169,224,32,64,15,169,0,32,64
150 DATA 15,174,2,241,16,251,96,72,169,32
160 DATA 32,64,15,104,73,255,73,8,141,0
170 DATA 241,72,104,72,104,234,73,8,141,0,241,96
180 REM ** INSTALL MACHINE LANGUAGE DRIVER PROGRAM AT HEX $0F00
190 FOR A=3840 TO 3921
200 READ B
210 POKE A,B
220 NEXT A
230 INPUT "NUMBER OF WORDS",N
240 N1=N
250 FOR A=1 TO N
260 PRINT "ENTER WORD #";A
270 INPUT W(A)
280 NEXT A
290 REM ** SET UP TO CLEAR SPEAK & SPELL
300 POKE 11,0
310 POKE 12,15
320 X=USR(Y)
330 REM ** SET UP TO TALK
340 POKE 11,15
350 FOR B=1 TO N*4 STEP 4
360 GOSUB 1000
370 S(B)=C(1)
380 S(B+1)=N
390 S(B+2)=C(3)
400 S(B+3)=C(2)
410 NEXT B
420 REM ** NOW GO AND TALK
430 L=3937
440 FOR C=1 TO N*4 STEP 4
450 FOR D=1 TO 4
460 POKE L+D,S(C+D-1)*16
470 NEXT D
480 X=USR(Y)
490 NEXT C
500 GOTO 390
1000  REM *CONVERT DECIMAL ADDRESS TO FOUR NYBBLES
1010  N=W((B+3)/4)
1020  IF N<0 THEN N=N+65536
1030  FOR E=1 TO 3
1040  C(E)=0
1050  NEXT E
1060  FOR R=3 TO 1 STEP -1
1070  IF N>=0 AND N<16^R THEN 1110
1080  C(R)=C(R)+1
1090  N=N-16^R
1100  GOTO 1070
1110  NEXT R
1120  RETURN
```

Listing 2. BASIC program (OSI's Microsoft 8K ROM BASIC) to control the Speak & Spell talking machine from a parallel port at hex memory locations $F100 (Output) and $F102 (Input). This program incorporates the machine-language driver in Listing 1.

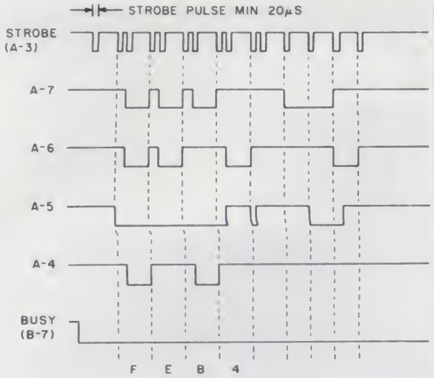


Fig. 2. Semidiagrammatic timing diagram of binary signals produced on each line of the parallel port by the machine-language driver of Listing 1. A-4 through A-7 and the Strobe line (A-3) are output lines of side A of the parallel port (see Fig. 1). The Busy line (B-7) is read by an input line on side B and remains high when the Speak & Spell cannot be accessed. As noted, the Strobe signal must remain low for at least 20 μs; however, the intervals between consecutive strobes do not appear to be very critical. The strobe intervals shown above are merely a function of the program logic of Listing 1 and the 1 MHz clock of my system. If you do encounter trouble (e.g., sporadic speech) with your own system, try adding NOPs (no operations) to increase the strobe width or to increase strobe interval times. The hex numbers at the bottom ($FEB4) represent the 16-bit address being sent on lines A-4 (LSB) through A-7 (MSB). This address will cause the word ENOUGH to be spoken.

instruction manual does not yet cover these machines, and my initial attempts at interfacing a Speak & Math to the Speak-2-Me-2 have only been partially successful.

Conclusion

In conclusion, I can fully recommend the Speak-2-Me-2 interface by the Percom Data Company (211 N. Kirby, Garland, TX 75042). The documentation is clear and complete for an easy connection to your TRS-80 machine, although for connection to another machine the instructions were a bit sketchy. However, I believe that as long as you can recreate the timing diagram of Fig. 2, you should have your machine talking back to you in no time, and at a very low cost.■

References

1. Wiggins, Richard, and Larry Brantingham, "Three-Chip System Synthesizes Human Speech," *Electronics* (Aug. 31, 1978), pp. 109–116.
2. Brantingham, Larry, "Speech Synthesis with Linear Predictive Coding," *Interface Age* (June 1979), pp. 72–75.

✓See List of Advertisers on page 210

Hop aboard Relativistic Spacelines and experience the mysteries of space travel.

An Incredible High-Speed Journey To the Stars

By Richard J. W. Hodgson

In an age when space travel excites the minds of young and old, it is a little strange that some of the restrictions facing future space travellers are not widely known or understood. I have found that simple computer programs can demonstrate some of these restrictions.

For example, astronauts who embark on high-speed trips to nearby stars will find that such journeys can profoundly affect their lives, especially when they return to Mother Earth. They will discover that they have aged less than those who have stayed behind.

This remarkable aging difference is described in Albert Einstein's theory of relativity. In view of the fact that we have celebrated Einstein's 100th birthday, it is worth examining this part of his theory, in a fashion which I hope will both stimulate and instruct.

The program depicts an imaginary space journey to a star. The operator is the passenger, and is treated in much the same way as we would imagine a person would be if he or she were checking in at the ticket counter of some future spaceline. In this case the carrier is called Relativistic Spacelines.

The passenger can also choose the spaceship's cruising speed. While this luxury may not be practical, it is important to study the advantages and disadvantages associated with travel at very high speeds.

The Twin Paradox

Before discussing the actual program, let's examine some of the background ideas on which it is based.

That aspect of Einstein's theory of relativity on which it is founded has historically been dubbed the "twin paradox." But there is really no paradox involved at all. It concerns the different rates of aging of two twins: one who completes a round trip space journey at speeds comparable to the speed of light, the other remaining behind on earth. Depending on how fast the traveling twin moves and how far away his journey takes him, when he returns to earth he will find that his earthbound twin is older than he is.

This difference in aging arises from a fundamental property of light discovered by Einstein. Unlike the behavior of objects with which we are familiar, light always travels at the same fixed speed, independent of the speed of the observer. This speed, usually denoted by the letter c, is about 186,000 miles, or 300,000 kilometers, per second.

If you are driving a car along the highway at 55 miles per hour, a car coming towards you at the same speed appears to approach at a speed of 110 miles per hour. This speed is referred to as the relative speed of one object as seen by another. The driver of the approaching car will observe the same phenomenon. In this case the relative speed is the sum of the individual speeds (actually it's the difference of the velocities). If instead you are driving at 55 mph and a car in front of you traveling in the same direction was moving at only 40 mph, then it appears as if you are approaching the second car at a speed of 15 mph. This same phenomenon is observed in a wide variety of situations.

Light is different. If a light wave were coming towards you and you were moving towards it at half of the speed of light ($c/2$), then the commonsense argument would be to say that the light appears to approach you at a speed of $c + c/2$, or $3c/2$. Not so, said Einstein. The light still appears to travel at the speed c, *independently of how fast you are moving*. This goes against all common sense, and is the reason that this was such a dramatic

Address correspondence to Richard J. W. Hodgson, 25 Downsview Cres., Ottawa, Ontario K2G 0A4, Canada.

theory. Yet at the time that Einstein proposed this theory, it was necessary to explain many of the observations that scientists had made.

The concept that the speed of light was a fundamental constant independent of the observer required some significant changes in the basic laws of physics. The speed of light is an upper limit: nothing can travel faster. In addition, one can no longer talk about simultaneous events occurring at the same time. What is simultaneous for one observer would not be simultaneous for another traveling at a different speed. Time was no longer something that was fixed. The time between two events for one observer would not be the same for another.

This particular aspect of the theory applies to our space travel problem. Furthermore, the faster an object travels, the more massive it becomes. That is, the mass of the object increases. The consequence is that the object becomes increasingly difficult to accelerate as its speed increases.

These ideas are not just theories on paper. They have been tested over and over again in the lab, and have always been verified. For example, there exist subatomic particles that can be accelerated very close to the speed of light. When this is done, the effects mentioned above are observed to occur. The mass of the particle does increase by the prescribed amount. The change of time scales can also be observed. Many of these subatomic particles have lifetimes; they exist only for a finite time, before decaying into something else. This lifetime can be measured when the particles are at rest and when they are moving at very high speeds. When the particles are moving at high speeds their lifetimes increase by the amount predicted by Einstein's theory.

The Program

The program to demonstrate these ideas is written in North Star BASIC. Wherever possible I have tried to avoid using statements peculiar to this interpreter, so that it may be easily adapted to other systems. Areas where changes may be required when using other interpreters are described below.

The "passenger" running this program is prompted to input five pieces of *information*. The first two are essential for the calculation, and the remaining three make the consequences more interesting.

First, the operator must choose the destination for his/her round-trip voyage. The traveler has a choice of ten stars, or can input the name of a different star. In the latter case the program asks for the name of the star, together with its distance in light-years. (A light-year is the distance a light beam travels in one year, approximately 5.9×10^{12} miles.) Note that the star distances stored in the DATA statement 1810 are in parsecs, a unit of distance commonly used in astronomy. One parsec is approximately 3.26 light-years.

The second piece of required information is the cruising speed, which is given as a percentage of the speed of light. Naturally the spaceship must be accelerated to this speed from earth, and *decelerated* at the end of the journey. I have assumed an acceleration of 5 g in line 670 of the program; that is, five times the acceleration of gravity at the earth's surface. This might appear high for comfort-

Sample run. The responses typed by the operator are in bold.

```
                    Welcome to
               Relativistic Spacelines

  Your name please. . . William
  Give me your date of birth (month,day,year)
  For example: 8,21,1940
Type date here: 10,17,1944

               Relativistic Spacelines

Passenger's name: William
Passenger's number: 656280

Input today's date.
Type date here: 1,10,1980

Give me the name of a friend or relative who will be remaining on earth. . .Judy
What is his/her birthdate?
Type date here: 5,21,1947

You may choose your destination from the following stars,
or by returning a 0, name your own star.
  1—Sirius              6—Procyon
  2—Canopus             7—Betelgeuse
  3—Alpha Centauri      8—Altair
  4—Vega                9—Spica
  5—Rigel              10—Antares

Type the number of the star you wish to visit
Here. . .4

The star Vega is 26.08 light-years away.
  Your spaceship to Vega can travel very close to the speed of light. In order to make your
trip enjoyable, you may choose your cruising speed as some percentage (less than 100) of
the speed of light.
For example, 80 or 95.
Input your speed, as a %, here: 85.0

               Relativistic Space-Lines
               End-of-journey Report

Return journey to Vega                   Passenger: William
Departure Date: Jan, 10, 1980            Distance: 26.08 light-years
Return date: Sep, 14, 2040               Speed factor: 85.00%
Passengers have aged                     32.9 years.
               60.7 years have passed on earth.

  Name       Age at departure     Age upon return
WILLIAM            35                  68
JUDY               32                  93

Do you want to take another trip? (Y/N) N
```

able travel, so if you're a bit squeamish you can reduce this. But be careful—if the acceleration is too slow, the spaceship may not have enough time to achieve its cruising speed before it has to start to brake. If this happens, the program prints a warning message giving the maximum speed attained.

The other three items of information required are dates—the birthdates of the space traveler and of a friend who is staying on earth, and the date of departure. These are used to determine the ages of both individuals upon return. I have not been rigorous in accounting for leap years when determining the date of return, but have only taken them into account in an average fashion. This makes the coding simpler, but at the same time doesn't detract from its effectiveness.

Modifications

If you want to run this program using a different BASIC, you'll need to make a number of modifications.

String arrays are treated differently in North Star BASIC than in most other interpreters. An array A$ dimensioned as A$(20) in North Star BASIC refers to a single string of at most 20 characters. A substring is then derived by writing A$(6,10), which refers to the substring consisting of characters 6 through 10 inclusive. This technique has been used here in storing the star names in a string "array" S$ which is read (line 590) from the two DATA statements at 1790 and 1800. Array C2( ), read from the last two DATA statements, is made up of integers referring to the first and last characters of each star name. Using these values the name of a particular star (a substring) is formed as in line 640. A similar procedure is employed for the months of the year in lines 1030–1050.

In several places the statement Print CHR$(11) is given. This clears the CRT screen in my system (a SOL or a VDM) and may have to be altered to perform the same function on your system, or else simply eliminated. The passenger number is obtained from a random number generator in line 280. The seed for the random number generator is set in line 140, and will have to be adjusted appropriately for your BASIC.

North Star BASIC also supports formatted output. The print statements in lines 290, 880, 1350 to 1370 and 1500 to 1510 contain terms, separated by commas, starting with a % sign. These describe either integer (I) or decimal (F) formatted output for the next character to be printed. In addition I have made use of the multiple statement per line facility wherein statements are separated by the ":" separator.

Conclusion

If you want to pursue the subject of relativity, your public library will have a number of easy-to-follow books. If you are interested in the mathematics of the present code, then look up the article by R. A. Muller in *The American Journal of Physics*, Vol. 40, p. 966, 1972. In any case you'll find this routine a valuable asset when it comes to demonstrating to your friends the superior "intelligence" and abilities of your computer—handling relativity calculations in a flash. Just don't let them know how simple the actual calculations are. ■

Program listing.

```
10 REM    ****************************************
20 REM    *   Relativistic Space Travel Program   *
30 REM   *                 by                    *
40 REM   *           R.J.W.  Hodgson             *
50 REM   *               Jan  1980               *
60 REM   ****************************************
70 REM
80 DIM A$(25),B$(25),C1$(36),M(12)
90 DIM C1(10),C2(20),S$(71),G$(15)
100 REM  CLEAR THE SCREEN TO START PROGRAM
110 PRINT CHR$(11)
120 PRINT
130 REM  THE NEXT LINE INITIALIZES THE RANDOM NO. GENERATOR
140 R=RND(-1)
150 PRINT TAB(27)," Welcome to"
160 PRINT TAB(21)," Relativistic Space-Lines"
170 PRINT \ PRINT \ PRINT \ PRINT
180 INPUT "  Your name please...",A$
190 PRINT "  Give me your date of birth (month,day,year)"
200 PRINT "  For example:  8,21,1940"
210 GOSUB 1720 \ REM  DATE INPUT SUBROUTINE
220 M1=M8 \ D1=D8 \ Y1=Y8
230 PRINT CHR$(11)
240 PRINT
250 PRINT TAB(21),"Relativistic Space-Lines"
260 PRINT
270 PRINT "Passenger's name:  ",A$
280 M = 1000000*RND(0) \ M=INT(M)
290 PRINT "Passenger number:  ",%7I,M
300 PRINT \ PRINT
310 PRINT "Input today's date."
320 GOSUB 1720
330 M2=M8 \ D2=D8 \ Y2=Y8
340 PRINT \ PRINT
350 PRINT "Give me the name of a friend or relative who will"
360 INPUT "be remaining on earth...",B$
370 PRINT "What is his/her birthdate?"
380 GOSUB 1720
390 M3=M8 \ D3=D8 \ Y3=Y8
400 REM  OBTAIN THE DESTINATION INFORMATION
410 PRINT CHR$(11)
420 PRINT
430 PRINT "You may choose your destination from the following"
440 PRINT "stars, or by returning a 0, name your own star."
450 PRINT "  1 - Sirius                   6 - Procyon"
460 PRINT "  2 - Canopus                  7 - Betelgeuse"
470 PRINT "  3 - Alpha Centauri           8 - Altair"
480 PRINT "  4 - Vega                     9 - Spica"
490 PRINT "  5 - Rigel                   10 - Antares"
500 PRINT
510 PRINT "Type the number of the star you wish to visit"
520 INPUT "Here...",S1
530 IF S1=0 THEN GOSUB 1570
540 IF S1<11 THEN 570
550 PRINT "Value of star number invalid.  Enter again."
560 GOTO 520
570 RESTORE
580 IF S1=0 THEN 650
590 READ S$(1,59) \ READ S$(60,71)
600 FOR I=1 TO 10 \ READ C1(I) \ NEXT I
610 FOR I=1 TO 20 \ READ C2(I) \ NEXT I
620 REM  C1 IS IN PARSECS,  D IS IN LIGHT YEARS.
630 D = 3.26*C1(S1) \ N=2*S1-1
640 G$=S$(C2(N),C2(N+1))
650 PRINT
660 PRINT "The star ",G$," is ",D," light years away."
670 G = 9.8 \ A = 5*G
680 REM  A = ACCELERATION OF THE SHIP.  MAY BE CHANGED.
690 PRINT "  Your spaceship to ",G$," can travel very close"
700 PRINT "to the speed of light.  In order to make your trip"
710 PRINT "enjoyable, you may choose your cruising speed as"
720 PRINT "some percentage (less than 100) of the speed of light."
730 PRINT "For example 80 or 95."
```

More

MICROPROCESSOR SUPPORT I.C.'S

WE GUARANTEE FACTORY PRIME PARTS

We are going to become the largest supplier of prime microprocessor support I.C.'S. We guarantee that our I.C.'S are purchased from manufacturer authorized distributors. This is the only way to deliver prime parts at the lowest possible prices. Our committment is to offer the best price and the fastest delivery to our customer. We give many thanks to our valued customers who have helped us grow.

NEC 16Kx1 DYNAMIC RAM 200 N.S.

These are prime 4116's from one of the best MOS RAM manufacturers in the world.

4116	200ns
8 for $25.00	32 for 96.00

8080A	CPU	4.95
8085A	CPU	8.95
8086	CPU	99.95
8088	CPU	44.95
Z-80	CPU	6.70
Z-80A	CPU	7.25

2708	EPROM 1Kx8	4.50
2716	EPROM 2Kx8	7.00
2732	EPROM 4Kx8	19.00
4118	STATIC 1Kx8	15.00
4164	200ns 64Kx1	Call
Z80B	CPU	21.00

NEC 1Kx4 STATIC RAM 250 N.S.

These are prime low power static ram's NEC for the finest in MOS MEMORY.

2114L	250ns
8 for $25.00	32 for $96.00

Z80-P10	6.00
Z80A-P10	7.10
Z80-CTC	6.00
Z80ACTC	7.10
Z80-DMA	18.50
Z80A-DMA	22.50
Z80-S10/0	18.50
Z80A-S10/0	22.50
Z80-S10/1	18.50
Z80A-S10/1	22.50
Z80-S10/2	18.50
Z80A-S10/2	22.50
3205	3.95
3242	10.00
8155	11.25
8185	29.95
8185-2	39.95
8202	45.00
8205	3.95
8212	2.00
8214	3.95
8216	1.85
8224	2.65
8226	1.85
8228	5.00
8238	5.45
8243	4.65
8251A	5.55
8253	9.85
8255A	5.40
8255A-5	5.40
8257	9.25
8257-5	9.25
8259A	7.30
8271	60.00
8275	32.95
8279	10.80
8279-5	10.80
8282	6.70
8283	6.70
8284	5.85
8286	6.70
8287	6.70
8288	25.40
8289	49.95

8755	49.95
4000	.35
4001	.35
4002	.35
4006	1.39
4007	.29
4008	1.39
4009	.49
4010	.49
4011	.35
4012	.29
4013	.49
4014	1.39
4015	1.15
4016	.59
4017	1.19
4018	.99
4019	.49
4020	1.19
4021	1.19
4022	1.15
4023	.38
4024	.79
4025	.38
4026	2.50
4027	.65
4028	.85
4029	1.29
4030	.45
4031	3.25
4032	2.15
4033	2.15
4034	3.25
4035	.95
4037	1.95
4040	1.29
4041	1.25
4042	.95
4043	.85
4044	.85
4046	1.75
4047	1.25
4048	.99
4049	.69

4050	.69
4051	1.10
4052	1.10
4053	1.10
4055	3.95
4056	2.95
4059	9.95
4060	1.39
4066	.75
4068	.35
4069	.35
4070	.49
4071	.35
4072	.35
4073	.35
4075	.35
4076	1.29
4078	.35
4081	.35
4082	.35
4085	1.95
4086	.79
4093	.99
4099	2.25
4104	1.99
4501	.39
4502	1.65
4503	.69
4505	8.95
4506	.75
4507	.95
4508	3.95
4510	1.39
4511	1.39
4512	1.39
4514	3.95
4515	3.95
4516	1.69
4519	.99
4520	1.39
4522	.99
4526	1.15
4527	1.75
4528	.99

4531	.99
4532	1.25
4539	.99
4543	1.99
4553	3.50
4555	.75
4556	.75
4581	1.99
4582	1.01
4584	.55
4585	.99
4702	9.95
74C00	.39
74C02	.39
74C04	.39
74C08	.49
74C10	.49
74C14	1.65
74C20	.39
74C30	.39
74C32	.99
74C42	1.85
74C48	2.39
74C73	.85
74C74	.85
74C85	2.49
74C89	4.95
74C90	1.85
74C93	1.85
74C95	1.85
74C107	1.19
74C151	2.49
74C154	3.50
74C157	2.10
74C160	2.39
74C161	2.30
74C163	2.39
74C164	2.39
74C173	2.59
74C174	2.75
74C175	2.75
74C192	2.39
74C193	2.39
74C195	2.39

74C925	6.95
74LS00	.35
74LS01	.28
74LS02	.28
74LS03	.28
74LS04	.39
74LS05	.28
74LS08	.39
74LS09	.39
74LS10	.28
74LS11	.39
74LS12	.39
74LS13	.47
74LS14	1.25
74LS15	.39
74LS20	.26
74LS21	.38
74LS22	.38
74LS26	.39
74LS27	.39
74LS28	.39
74LS30	.26
74LS32	.39
74LS37	.79
74LS38	.39
74LS42	.79
74LS47	.79
74LS48	.79
74LS51	.26
74LS54	.35
74LS55	.35
74LS73	.45
74LS74	.59
74LS75	.68
74LS76	.45
74LS78	.65
74LS83	.99
74LS85	1.19
74LS86	.45
74LS90	.75
74LS92	.75
74LS93	.75
74LS95	.88
74LS96	.98

74LS107	.45
74LS109	.45
74LS112	.49
74LS122	.55
74LS123	1.19
74LS125	1.35
74LS126	.89
74LS132	.79
74LS136	.59
74LS138	.89
74LS139	.89
74LS145	1.25
74LS148	1.49
74LS151	.79
74LS153	.79
74LS155	1.19
74LS156	.99
74LS157	.99
74LS158	.75
74LS160	.98
74LS161	1.15
74LS162	.98
74LS163	.98
74LS164	1.19
74LS165	.89
74LS166	2.49
74LS170	1.99
74LS173	.89
74LS174	.99
74LS175	.99
74LS181	2.20
74LS190	1.15
74LS191	1.15
74LS192	.98
74LS193	.98
74LS194	1.15
74LS195	.95
74LS196	.89
74LS197	.89
74LS221	1.49
74LS240	1.95
74LS241	1.90
74LS242	1.95
74LS243	1.95

74LS244	1.95
74LS245	4.95
74LS247	1.10
74LS248	1.10
74LS249	1.69
74LS251	1.79
74LS253	.98
74LS257	.98
74LS258	.98
74LS259	2.95
74LS260	.69
74LS261	2.49
74LS266	.59
74LS273	1.75
74LS275	4.40
74LS279	.59
74LS283	1.10
74LS290	1.29
74LS293	1.95
74LS295	1.10
74LS298	1.29
74LS324	1.75
74LS347	1.95
74LS348	1.95
74LS352	1.65
74LS353	1.65
74LS363	1.49
74LS365	.99
74LS366	.99
74LS367	.73
74LS368	.73
74LS373	2.75
74LS374	2.75
74LS375	.69
74LS377	1.95
74LS385	1.95
74LS386	.65
74LS390	1.95
74LS393	1.95
74LS395	1.70
74LS399	2.95
74LS424	2.95
74LS668	1.75
74LS670	2.29

MAIL ORDERS SHOULD BE SENT TO:
13400 Northup Way 22 Bellevue Washington 98005
Telephone Orders & Inquiries 1-800-426-2668
Minimum Order $10.00 Add $3.00 Shipping

HANLEY ENGINEERING
(206) 643-0792
13400 NORTHUP WAY 22
BELLEVUE, WASHINGTON 98005

FOR THE FINEST IN MICROPROCESSOR SUPPORT I.C.'S

Listing continued.

```
740 INPUT "Input your speed, as a %, here: ",V
750 IF V<100 THEN 790
760 PRINT " Sorry, we cannot travel faster than the speed of"
770 PRINT "light.  Try again."
780 GOTO 740
790 REM  CALCULATE THE DISTANCE NEEDED TO ACHIEVE V.
800 V=V/100
810 G4=SQRT(1-V↑2) \ D4=9.5064*(1/G4-1)/A
820 IF 2*D4<D THEN 920
830 REM  FINAL SPEED WILL NOT BE ACHIEVED
840 PRINT CHR$(11)
850 PRINT "Your selected cruising speed cannot be achieved."
860 V1=SQRT(1-1/(1+A*D/19.013)↑2)
870 V2 = 100*V1
880 PRINT "Your maximum speed will be ",%5F2,V2,"% that of light."
890 V=V1 \ L=0
900 PRINT \ PRINT \ INPUT "Press RETURN to continue...",R$
910 GOTO 940
920 L = D - 2*D4
930 REM  NOW COMPUTE THE TIMES
940 T = 4*V/(A*G4)+2*L/V
950 T1 = (19.013/A)*LOG((1+V)/(1-V))+2*L*G4/V
960 T2 = T-T1
970 PRINT CHR$(11) \ PRINT
980 PRINT TAB(21),"Relativistic Space-Lines"
990 PRINT TAB(23),"End-of-Journey Report"
1000 PRINT "-------------------------------------------------------"
1010 PRINT "Return journey to ",G$,TAB(33),"Passenger:",A$
1020 REM  NOW COMPUTE THE DATES
1030 C1$="JanFebMarAprMayJunJulAugSepOctNovDec"
1040 I1=3*M2-2 \ I2=I1+2
1050 M$=C1$(I1,I2)
1060 PRINT "Departure Date: ",M$,",",D2,",",Y2,TAB(33),"Distance: ",D," light
 yrs."
1070 A1=Y2-Y1 \ IF M2<M1 THEN A1=A1-1
1080 IF M2<>M1 THEN 1100
1090 IF D2<D1 THEN A1=A1-1
1100 A4=A1
1110 A2=Y2-Y3 \ IF M2<M3 THEN A2=A2-1
1120 IF M2<>M3 THEN 1140
1130 IF D2<D3 THEN A2=A2-1
1140 A5=A2
1150 Y9=INT(T) \ D9=INT(365.25*(T-Y9))
1160 IF M2>2 THEN 1180
1170 F=M2+13 \ G=Y2-1 \ GOTO 1190
1180 F=M2+1 \ G=Y2
1190 N6=INT(365.25*(G-1900))+INT(30.6*F)+D2
1200 REM N6 = DEPARTURE DATE
1210 G=Y2 \ F=13
1220 N7=INT(365.25*(G-1900))+INT(30.6*F)+31
1230 D8=N7-N6 \ REM D8= DAYS REMAINING IN YEAR
1240 IF D9>D8 THEN 1300
1250 G=Y2-1
1260 N7=INT(365.25*(G-1900))+INT(30.6*F)+31
1270 D7=N6-N7 \ D9=D7+D9
1280 GOSUB 1620
1290 GOTO 1320
1300 Y9=Y9+1 \ D9=D9-D8
1310 GOSUB 1620
1320 Y9=Y2+Y9 \ I1=3*M9-2 \ I2=I1+2
1330 V9=100*V
1340 PRINT "Return date: ",C1$(I1,I2),",",D9,",",Y9,
1350 PRINT TAB(33),"Speed factor: ",%5F2,V9,"%"
1360 PRINT TAB(6),"Passengers have aged ",%10F1,T1," years."
1370 PRINT TAB(6),%10F1,T," years have passed on earth."
1380 PRINT
1390 REM COMPUTE AGE OF TRAVELLER
1400 Y8=INT(T1) \ D8=INT(365.25*(T1-Y8))
1410 A1=A1+Y8
1420 IF D8>182 THEN A1=A1+1
1430 A2=A2+INT(T)
1440 G$="    "
1450 F$="    "
1460 IF A1>100 THEN F$="!!!!"
1470 IF A2>100 THEN G$="!!!!"
1480 IF INT(365*(T-INT(T)))>182 THEN A2=A2+1
1490 PRINT " Name                Age at departure      Age upon return"
1500 PRINT A$,TAB(20),%5I,A4,TAB(40),A1," ",F$
1510 PRINT B$,TAB(20),%5I,A5,TAB(40),A2," ",G$
1520 PRINT \ PRINT
1530 INPUT " Do you want to take another trip? (Y/N)",A$
1540 IF A$(1)="Y" THEN 110
1550 STOP
1560 REM ROUTINE TO ACCEPT STAR NAME AND DISTANCE.
1570 PRINT \ PRINT
1580 INPUT "What is the name of the star? ",G$
1590 INPUT "It's distance from the earth in light-years?",D
1600 RETURN
1610 REM ROUTINE TO DETERMINE THE MONTH AND THE DAY.
1620 FOR I=1 TO 12
1630 M(I)=31 \ NEXT I
1640 M(2)=28 \ M(4)=30 \ M(6)=30 \ M(9)=30 \ M(11)=30
1650 FOR I=1 TO 12
1660 D5=D9 \ D9=D9-M(I)
1670 IF D9<=0 THEN EXIT 1690
1680 NEXT I
1690 M9=I \ D9=D5
1700 RETURN
```

Maxi Manager

The finest Data Base Manager Available

DATA MANAGEMENT PROGRAM COMPARISON CHART

	CCA DATA MANAGER	AIDS III with CALS	MAXI MANAGER	RADEX 10	PROFILE
FILE CAPACITY & FORMAT					
Maximum # of disks per file	1	1	4	31	4
Maximum # of records per file	2450	Note 1	32,767	10,199	65,535
Maximum record length	249	254	800	255	255
Maximum # of characters per field	249	254	40	254	255
Maximum # of fields	24	20	20	127	153
Maximum # of characters per field label	15	10	19	12	765
Variable length records (pack sectors)	No	Note 2	Yes	No	No
FIELD TYPES					
Alphanumeric	Yes	Yes	Yes	Yes	Yes
Numeric	Yes	Yes	Yes	Yes	No
Fixed decimal numeric	Note 4	Yes	Yes	No	No
Date (MM/DD/YY)	Yes	No	Yes	No	No
Extended date (MM DD YYYY)	No	No	Yes	No	No
Calculated equation	Note 5	Note 6	Yes	No	No
Permanent fields	Yes	No	No	No	No
SORTING					
Machine language assisted	No	Yes	Yes	Note 7	Yes
Sort by any field	Yes	Yes	Yes		Yes
Number of Sort Key files	1	1	5		1
Numeric sort	Yes	Yes	Yes		No
Ascending sort	Yes	Yes	Yes		Yes
Descending sort	Yes	Yes	Note 11		Yes
Sort within a selected range	No	No	Yes		No
Sort multiple fields simultaneously	Yes	Yes	No		No
FILE MAINTENANCE					
Fixed length input fields	Yes	Yes	Yes	Yes	Yes
Single key entry of common data	No	No	Yes	No	No
Single field EDIT selection	Yes	Yes	Yes	Yes	Yes
Skip record (next or previous)	Yes	Yes	Yes	No	Yes
Search & EDIT record	No	Yes	Yes	No	Yes
Search & DELETE record	No	Yes	Yes	No	No
Auto rejection of alphanumeric data in numeric field	Yes	No	Yes	No	No
RECORD SELECTION TECHNIQUES					
Record number	Yes	Yes	Yes	Yes	No
Binary search (high speed)	No	No	Yes	No	No
Maximum # of simultaneous keys	1	4	10	31	1
RELATIONAL COMPARISONS					
Equal	No	Yes	Yes	Yes	Yes
Not equal	No	Yes	Yes	No	Yes
Greater than	No	Yes	Yes	Yes	Yes
Less than	No	Yes	Yes	Yes	Yes
Instring	Yes	No	Yes	Yes	No
AND / OR	No	No	Yes	Yes	No
Wild card masking	No	No	Yes	No	No
PRINTING					
User specified page title	Note 8	Yes	Yes	No	Note 10
User specified column headings	No	No	Yes	No	Yes
Automatic page numbering	Yes	Yes	Yes	Yes	Yes
Right justification	No	Yes	Yes	No	No
User defined column widths	Yes	No	Yes	Yes	Yes
User defined column separators	No	No	Yes	No	No
Keyboard entered columnar values	No	No	Yes	No	No
Merge data into form letters	No	No	Yes	No	No
Form filling applications	No	No	Yes	No	No
Columnar totals	Yes	Yes	Yes	No	No
Columnar subtotals generated upon change in a specific field	Yes	Yes	Yes	No	No
Built in screen print	No	No	Yes	No	No
MISCELLANEOUS					
Cost	$75.00	$94.90	$99.95	$99.00	$79.95
Punctuation allowed within data fields	Yes	?	Yes	Yes	Yes
Upper / Lower case	Note 3	Note 3	Yes	Note 3	Note 3
Built in RS-232-C driver	Note 3	Note 3	Yes	Note 3	Note 3
Built-in TRS-232 driver	Note 3	Note 3	Yes	Note 3	Note 3
Programmer's interface	Note 9	Note 9	Yes	No	Note 9
Sample DATA disk	No	No	Yes	No	No
Documentation (# of pages)	?	?	120	38	29

NOTE 1 File size is dependent on memory size.
NOTE 2 Sequential files only.
NOTE 3 User must apply own driver routine.
NOTE 4 Hard copy print out only
NOTE 5 Four functions (+ - * /) only
NOTE 6 Same as note #5 with a maximum of two calculated fields.
NOTE 7 Available as a separate program for $99 95.
NOTE 8 120 character maximum.
NOTE 9 Data structures defined in manual.
NOTE 10 132 characters maximum.
NOTE 11 User option (files can be read from ascending or descending order).

The jury is in and the verdict is . . . "outstanding!" Reviews from all of you who purchased MAXI MANAGER (not to mention raves by many top microcomputing magazines) have heralded it as the definitive data base managing system. We knew that business owners and hobbyists demanded the finest data base managing system available. To all of you who praised us for MAXI MANAGER, we extend our thanks. And to those of you who have yet to try MAXI MANAGER, we invite you to experience this incredible system today. But don't take our word for it (or our jury's). Judge for yourself.

JUST CHECK SOME OF THESE FEATURES

- Supports six different relational search techniques.
- Comes with programmer's interface.
- Over 120 pages of documentation.
- Supports up to 20 user defined fields of 40 characters each.
- Record length up to 800 characters.
- Files can be up to four disks in length.
- Compatable 35, 40, 77 & 80 track drives with proper operating system.
- Has calculated equation fields.
- Complete report generator.
- Works hand in hand with any word processor.

MODEL 1 version requires TRSDOS 2.3 and is compatable with NEWDOS 2.1 & NEWDOS 80

MODEL 3 version comes on TDOS, a special version of the DOSPLUS operating system.

Requires 48K of RAM and one disk drive minimum.

For the TRS-80 Model 1 & 3
Order No. 012-0096

$99.95

A Division of Scott Adams, Inc.
Adventure International • Box 3435 • Longwood, FL 32750
TOLL-FREE ORDER NUMBER: (800) 327-7172
IN FLORIDA (305) 862-6917
SHIPPING AND HANDLING CHARGES ARE EXTRA
PRICES ARE SUBJECT TO CHANGE WITHOUT NOTICE

✓220

ai

This North Star program provides a simple, inexpensive way to write letters on your microcomputer.

El Cheapo Word Processing

By John E. Hafey

Since most word processing programs are expensive and require special hardware configurations that I don't have, I needed an inexpensive way to write letters using my computer as an electric typewriter.

This program, written in North Star BASIC, Version 6, has nine options: you can write/save or write/not save your letters and add to an existing letter, correct a line, insert a line, delete a line, review a letter, destroy a letter and print any letter saved on disk or as many copies of it as you wish. The program also provides a listing of existing letters on the disk at the beginning of a run. I borrowed this feature from the Directory Listing program published in *Kilobaud Microcomputing* (October 1979, p. 116).

To use this feature, you must create a file called "★★★★★★★★". Enter the following on the disk: CR ★★★★★★★★ 4 0 and then TY ★★★★★★★★ 3. This does not use any normal user file space, but occupies the same space as the disk directory. If you do not wish to use this feature, then eliminate lines 170 to 420, as well as lines 70 and 120.

Program Features

The program is self explanatory and will take you through it without any difficulty. It will run on a 16×64 video display. I use the Heathkit H-14 printer for hard copy. But in case you want to make changes to meet your hardware requirements, let's take a closer look at the program.

The parameters for the computer are as follows. Line 70 sets the number of non-type-3 files on your disk. Line 80 sets the line length or characters per line. Line 90 sets the page size or number of lines per page and line 100 sets the maximum number of pages per letter.

I ignore the prompt command to turn on the printer when in the write/save mode and write the letter by video display. Then I can review the letter for errors before printing.

When correcting a line, you have to retype the entire line. Without a programmable cursor you can't just correct a letter or word.

The printer is set to print 80 characters per line, six lines to the inch. The H variable is the on-off function. If you don't have the H-14 printer, then change lines 2400–2420 to the configuration your printer needs.

Also, when printing your letters, the print head will travel the full width of the page. This is due to the fact that each line is one complete record in the file and contains both printing characters and blanks. This was needed in order to be able to insert, delete and correct individual lines in the letter.■

John E. Hafey, 5636 Drake Ave., Cleveland, OH 44127.

Word processing program in North Star BASIC.

```
10REM*  Word Processing Program    Ver. 3.1  2/15/81
20REM*
30REM*  Written by John Hafey, 5636 Drake Ave.
40REM*                Cleveland, Ohio     44127
50REM*
60REM*
70A=03\REM*  'A' Equals number of non type 3 files on disk
80B=60\REM*  'B' Sets line length (chrs per line)
90C=53\REM*  'C' Sets page size (lines per page)
100D=03\REM*  'D' Sets MAXIMUM number of pages per letter
110REM*
120DIMX$(16),D$(1024)
130DIMA$(B),B$(B),C$(B),E$(B),F$(B),U$(B+5)
```

More

Listing continued.

```
140FORJ=1TOB\F$(J,J)=" "\NEXTJ\S=B+2
150E=INT(S*C/256)+1\REM* 'E' SETS FILE SIZE
160GOSUB2300\!TAB(18),"-*- WORD PROCESSING PROGRAM -*-"\!
170REM*
180REM* DISK QUERY ROUTINE
190REM*
200OPEN#0,"********"\O=1\I=0
210READ#0,&T
220IFT=32THEN300
230X$="        "
240FORJ=1TO8\X$(J,J)=CHR$(T)\READ#0,&T\NEXTJ
250IFX$<>"********"THEN270
260O=O-1\GOTO310
270U=FNA(O)\D$(U,U+7)=X$\D$(U+8,U+8)=CHR$(T)
280FORJ=U+9TOU+15\READ#0,&T\D$(J,J)=CHR$(T)\NEXTJ
290O=O+1\I=I+1\IFI<64THEN210ELSE330
300READ#0,&T,&T,&T,&T,&T,&T,&T
310READ#0,&T,&T,&T,&T,&T,&T,&T,&T
320I=I+1\IFI<64THEN210
330CLOSE#0\IFO>ATHEN350
340!"No ACTIVE entries on file!!!"\GOTO460
350!"List of entries on file:",\X=1
360FORI=1TOO*16STEP16
370V=ASC(D$(I+12))\IFV<3THEN420
380IFX/2=INT(X/2)THEN400
390!TAB(30),D$(I,I+7),\GOTO410
400!TAB(45),D$(I,I+7)
410X=X+1
420NEXTI
430REM*
440REM* MENU ROUTINE
450REM*
460!\!"The following OPTIONS are available:"\!
470!TAB(2),"1. Write/Don't Save",TAB(24),"4. Add to Letter",
480!TAB(44),"7. Delete a Line"
490!TAB(2),"2. Write/Save",TAB(24),"5. Correct a Line",
500!TAB(44),"8. Print a Letter"
510!TAB(2),"3. Review a Letter",TAB(24),"6. Insert a Line",
520!TAB(44),"9. Destroy a Letter"
530!TAB(43),"10. Exit program"
540INPUT"Which Option: ",Y\!
550IFY<1ORY>10THEN540
560G=1\K=0\F=1\H=0\N=0\P=0
570IFY=1THEN640\IFY=2THEN1020\IFY=3THEN1100
580IFY=4THEN1190\IFY=5THEN1310\IFY=6THEN1470
590IFY=7THEN1710\IFY=8THEN1940\IFY=9THEN2200
600IFY=10THEN2730
610REM*
620REM* WRITE ROUTINE
630REM*
640GOSUB2430
650INPUT"Single or Double space (1 or 2)? ",F\!
660IFF<1ORF>2THEN650
670GOSUB2300
680!"     At the ':' symbol, write a line of text.  You are"
690!"limited to",B," characters per line.  If you write more"
700!"than",B," characters, you will have to re-enter that"
710!"line again."\!
720IFN=0THEN760
730!"     You may leave the Printer off if you wish to Edit"
740!"the letter before actually printing the FINAL version."\!
750IFN=1THEN790
760!"CAUTION:  Printer must be ON when using this Option,"
770!"          as a line of text will be Printed when you"
780!"          press RETURN!!!"\!
790GOSUB2400
800!"Use the '↑' to END the letter you are writting.\!\!
810IFN=0THEN830
820OPEN#0,C$
830FORI=1TOP*C
840GOSUB2470\B$=U$
850IFB$="↑"THEN950
860IFF=1THEN880
870!#H
880!#H,B$\Z=Z+1
890IFZ<>INT(C/F)THEN910
900GOSUB2490
910IFN=0THEN930
920B$=B$+F$\WRITE#0,B$
930NEXTI
940!\!"THE FILE IS FULL!!!"\!
950IFN=0THEN970
960CLOSE#0
970!#H,CHR$(12)\Z=0
980!\GOTO2690
990 REM*
1000 REM* SAVE ROUTINE
```

More →

✓See List of Advertisers on page 210

Listing continued.

```
1010 REM*
1020 INPUT"Name of letter to be SAVED (8 chars)....",C$\!
1030 GOSUB2360\IFFILE(C$)<>3THEN1050
1040 !"That is a CURRENT data file, Re-enter"\GOTO1020
1050 GOSUB2430\CREATEC$,P*E
1060 N=1\GOTO670
1070REM*
1080REM*  REVIEW ROUTINE
1090REM*
1100INPUT"Name of letter to REVIEW....",C$\!
1110GOSUB2360\IFFILE(C$)=3THEN1130
1120GOSUB2370\GOTO1100
1130GOSUB2300\GOSUB2510\GOSUB2320
1140INPUT"Review another letter? ",T$\!
1150IFT$="Y"THEN1100ELSE2690
1160REM*
1170REM*  ADD ROUTINE
1180REM*
1190INPUT"Name of letter to ADD to....",C$\!
1200GOSUB2360\IFFILE(C$)=3THEN1220
1210GOSUB2370\GOTO1190
1220GOSUB2300\OPEN#0,C$,M
1230IFTYP(0)=0THEN1260
1240READ#0,B$\!"  ",B$
1250GOTO1230
1260P=M/E\N=1
1270H=1\GOTO830
1280REM*
1290REM* CORRECT ROUTINE
1300REM*
1310INPUT"Name of letter to CORRECT....",C$\!
1320GOSUB2360\IFFILE(C$)=3THEN1340
1330GOSUB2370\GOTO1310
1340K=1\GOSUB2300\GOSUB2510\GOSUB2310
1350INPUT"Correct which line? ",L\!
1360IFL=9999THEN1430
1370!"See OLD line, enter NEW line below...."\!
1380READ#0%(L-1)*S,B$\!": ",B$
1390GOSUB2470\!\B$=U$\B$=B$+F$
1400WRITE#0%(L-1)*S,B$,NOENDMARK
1410INPUT"Correct another line? ",T$\!
1420IFT$="Y"THEN1350
1430CLOSE#0\GOTO2690
1440REM*
1450REM*  INSERT ROUTINE
1460REM*
1470INPUT"Name of letter to INSERT line....",C$\!
1480GOSUB2360\IFFILE(C$)=3THEN1500
1490GOSUB2370\GOTO1470
1500K=1\GOSUB2300\GOSUB2510\GOSUB2310
1510INPUT"Insert before which line (by number)....",L\!
1520IFL=9999THEN1670
1530!"Now enter the line you wish to insert...."\!
1540GOSUB2470\B$=U$\B$=B$+F$
1550WRITE#0,B$
1560FORI=QTOLSTEP-1
1570READ#0%I*S,A$
1580READ#0%(I-1)*S,B$
1590WRITE#0%I*S,B$,NOENDMARK
1600WRITE#0%(I-1)*S,A$,NOENDMARK
1610NEXTI
1620READ#0%Q*S,E$
1630WRITE#0%Q*S,E$
1640!\INPUT"Insert another line? ",T$\!
1650IFT$<>"Y"THEN1670
1660Q=Q+1\GOTO1510
1670CLOSE#0\GOTO2690
1680REM*
1690REM*  DELETE ROUTINE
1700REM*
1710INPUT"Name of letter to DELETE from....",C$\!
1720GOSUB2360\IFFILE(C$)=3THEN1740
1730GOSUB2370\GOTO1710
1740K=1\GOSUB2300\GOSUB2510\GOSUB2310
1750INPUT"Delete which line (by number)....",L\!
1760IFL=9999THEN1900\L=L-1
1770READ#0%L*S,B$\!B$\!
1780INPUT"Delete this line? ",T$\!
1790IFT$<>"Y"THEN1880
1800Q=Q-1\READ#0%Q*S,B$
1810FORJ=(Q-1)TOLSTEP-1
1820READ#0%J*S,A$
1830WRITE#0%J*S,B$,NOENDMARK
1840B$=A$
1850NEXTJ
1860READ#0%(Q-1)*S,E$
1870WRITE#0%(Q-1)*S,E$
```

More

Listing continued.

```
1880INPUT"Delete another line? ",T$\!
1890IFT$="Y"THEN1750
1900CLOSE#0\GOTO2690
1910REM*
1920REM*  PRINT ROUTINE
1930REM*
1940INPUT"Name of letter to be PRINTED....",C$\!
1950GOSUB2360\IFFILE(C$)=3THEN1970
1960GOSUB2370\GOTO1940
1970!"How many copies of '",C$,"' ",
1980INPUT"letter? ",L\!
1990INPUT"Single or Double space (1 or 2)? ",F\!
2000IFF<1ORF>2THEN1990
2010GOSUB2400
2020FORJ=1TOL
2030OPEN#0,C$,M
2040FORI=1TO(M/E)*C
2050IFTYP(0)=0THENEXIT2130
2060READ#0,B$
2070IFF=1THEN2090
2080!#H
2090!#H,B$\Z=Z+1
2100IFZ<>INT(C/F)THEN2120
2110GOSUB2490
2120NEXTI
2130!#H,CHR$(12)\Z=0
2140CLOSE#0
2150G=1\NEXTJ
2160GOTO2690
2170REM*
2180REM*  DESTROY ROUTINE
2190REM*
2200INPUT"Name of letter to DESTROY....",C$\!
2210GOSUB2360\IFFILE(C$)=3THEN2230
2220GOSUB2370\GOTO2200
2230GOSUB2300\GOSUB2510
2240!"Sure you want to ",\GOSUB2320
2250GOTO2690
2260REM*
2270REM*  PROGRAM SUBROUTINES
2280REM*
2290DEFFNA(W)\W=(W-1)*16+1\RETURNW
2300FORJ=1TO16\!\NEXTJ\RETURN
2310!"(Enter 9999 to exit routine)"\RETURN
2320INPUT"Destroy this letter? ",T$\!
2330IFT$<>"Y"THENRETURN
2340!"Enter '↑' to destroy letter....",\T$=INCHAR$(0)
2350!\!\IFT$<>"↑"THENRETURN\DESTROYC$\RETURN
2360IFC$="↑"THEN2710ELSERETURN
2370!"That letter does NOT exist...."
2380!"Enter '↑' to restart program or"
2390!"Re-enter ",\RETURN
2400IFH=1THENRETURN
2410INPUT"Turn printer on....Type RETURN: ",T$\!
2420!#1,CHR$(27),CHR$(120),CHR$(1)\H=1\RETURN
2430!"Number of pages (",D,\INPUT" Max.)? ",P\!
2440IFP<=DTHENRETURN
2450!"1 to",D," pages only,"
2460!"Re-enter ",\GOTO2430
2470INPUT": ",U$\IFLEN(U$)<=BTHENRETURN
2480!"Line length, re-enter"\GOTO2470
2490!#H,CHR$(12)\Z=0\G=G+1
2500!#H,TAB(30),"--",G," --"\!#H\!#H\RETURN
2510OPEN#0,C$,M\R=0\Q=0
2520FORI=1TO(M/E)*C
2530IFTYP(0)=0THENEXIT2660
2540READ#0,B$
2550R=R+1\Q=Q+1
2560IFT$<>"↑"THEN2580
2570GOTO2530
2580!%2I,I," ",B$
2590IFR<>14THEN2650
2600IFK<>1THEN2640
2610!\!"Use '↑' when line located, else RETURN: "
2620T$=INCHAR$(0)\!\IFT$="↑"THEN2530
2630GOSUB2300\R=0\GOTO2650
2640!\INPUT"Type RETURN to continue....",T$\!\R=0
2650NEXTI
2660IFK=1THEN2680
2670CLOSE#0\!\RETURN
2680!\RETURN
2690INPUT"Do you wish to run program again? ",T$\!\!
2700IFT$<>"Y"THEN2730
2710GOSUB2300\FORJ=1TO1024STEP16
2720D$(J)="                    "\NEXTJ\GOTO200
2730END
```

When you find yourself in a tight corner, plan your escape with these simple circuits.

A Shortcut through the Gates

By Howard Rifkin

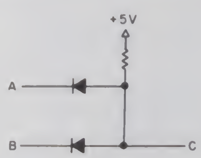


Fig. 1. Improvised AND gate.

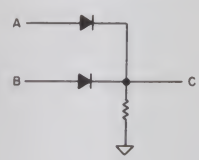


Fig. 2. OR gate.

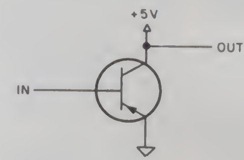


Fig. 3. NOT gate.

A few years ago I was working on a circuit which just fit into the prototype area of my 1802 evaluation kit, when I discovered that I needed one more AND gate. There was no room left to put a 7408 quad AND IC, and it seemed a waste to use a four-gate chip when only one gate was needed. I didn't have a 7408 anyway, so I was temporarily stuck.

I mentioned the problem to an engineer at work, who suggested the circuit in Fig. 1. As you can see, this makes a compact AND gate; if inputs A or B are zero, the appropriate diode will conduct and point C will be zero. If both inputs are 1, neither diode will conduct and the pull-up resistor will make point C 1.

I recently had a similar experience. I needed a single OR gate, and, remembering how I made the AND, I came up with the circuit in Fig. 2. In this case, if either input is 1, its diode will conduct, making the output 1. If both inputs are 0, the pull-down resistor will make the output 0.

To complete this logic technology, we need a NOT gate, but this can't be done with just diodes. However, we can get our NOT gate with the use of a transistor, as shown in Fig. 3.

It is important to note that these diodes and transistors will not substitute for gates in some computer circuits, because their current drain is too high. While I would not throw away my *TTL Cookbook*, these three circuits do work for me when I need one more gate in that crowded area. ■

Address correspondence to Howard Rifkin, 8 Harold St., Old Bridge, NJ 08857.

✓See List of Advertisers on page 210

DEALER DIRECTORY

El Monte, CA

Ohio Scientific specialist in the San Gabriel Valley serving greater Los Angeles. Full product line on display. Specializing in business computers. In-house service. Custom programming. Terminals. Printers. Open Mon–Sat, 9 AM–7 PM. **Computer & Video, 3380 Flair Dr., Suite 207, El Monte, CA 91731, 572-7292.**

Riverside, CA

Visit our Computer Support Center for the Inland Empire's largest selection of ICs, books and computer accessories. Open daily. Check our prices and friendly service. **Inland Electro-Mart, 8624 California Ave., Riverside, CA 92504, 687-3776.**

San Jose, CA

New and used computer products—specializing in S-100 boards, printers, drives, chassis and complete systems, as well as supplies and parts—Imsai, Tandon, Diablo—5000 sq. ft. **W/W Component Supply, Inc., 1771 Junction Ave., San Jose, CA 95112.**

Nokomis, FL

We are the leading area computer store. We carry Cromemco, Apple, Vector Graphic; printers & terminals. We offer full software support including G/L, A/R, payroll & word processing. **Computer Centre, 909 S. Tamiami Trail, Nokomis, FL 33555.**

Mention Kilobaud Microcomputing when you shop!

Sarasota, FL

Your personal and business computer store for Dynabyte, Vector, HP-85, Atari and Epson. Structured Systems and Micro-Pro software. Computer furniture and books by Osborne or Hayden. Sales, service and supplies. **Computer Crossroads, 3800 S. Tamiami Trail, Sarasota, FL 33579, 349-0200.**

Tampa, FL

Apple Computer sales and service. S-100 boards from SSM, Godbout, Thinker Toys, California Computer Systems. Computer books and magazines. **AMF Microcomputer Center, Inc., 11158 N. 30th Street, Tampa, FL 33612, 971-4072, 977-0708.**

Aurora, IL

Microcomputer systems for home or business; peripherals, software, books & magazines. Apple, Hewlett-Packard Series 80 Systems, HP Calculators, IDS, Qume, Starwriter printers. **Farnsworth Computer Center, 1891 N. Farnsworth Ave., Aurora, IL 60505, 851-3888.**

Chicago, IL

Computer Hardware/Specialists for home and business. Largest selection of computer books, magazines and copyrighted software in Illinois. Experienced factory-trained service department. Feature Apple, Alpha Microsystems and Hewlett-Packard calculators and accessories. **Data Domain of Schaumburg, 1612 E. Algonquin Rd., Schaumburg, IL 60195, 397-8700.**

Herington, KS

Hardware support. Maintenance and service for all microcomputers and peripherals. Kits assembled or debugged. Radio Shack (mods OK) repaired. Quality work, fast turnaround and reasonable cost. **Prairie Micro Clinic, Box 325, Herington, KS 67449, 258-2179.**

Garden City, MI

Books, magazines, hardware and software for Apple, North Star, TRS-80 and PET. **Computer Center, 28251 Ford Rd., Garden City, MI 48135, 425-2470.**

River Edge, NJ

Discount software—up to 25% off business, utility, recreational, educational and home programs. Apple, Atari, TRS-80 and PET. Atari computers always on sale. **Software City, 111 Grand Ave., River Edge, NJ 07661.**

Chautauqua, NY

Retail book store featuring the Disassembled Handbook for TRS-80 Volumes 1, 2, 3. English, German & French language editions. 9 AM–5 PM weekdays. Come and visit us. **Richcraft Computer Book Store, 1 Wahmeda Ave., Chautauqua, NY 14722, 753-2654.**

Portland, OR

Ohio Scientific specialists for business and personal computers. Local service. Terminals, printers, custom programming. Full OSI product line on display! 10 AM to 6 PM M–F. **Fial Computer, 11266 SE 21st Ave., Milwaukie, OR 97222, 654-9574.**

Montreal, Quebec

We do expert service on all microcomputers and peripherals, (CRT, printer, floppy disk) North Star, Hazeltine, Cromemco, Centronics, Shugart, Siemens, Apple, TRS, Epson, S-100. **Montreal Data Centre, 120 Ricard, Legardeur, Montreal, Quebec, 585-8801.**

Alexandria, VA

Full service center for Heath/Zenith computer products. Hardware, software on display and available off the shelf. Call or come by for a personalized demonstration. **Heathkit Electronic Center, 6201 Richmond Highway, Alexandria, VA 22303, 765-5515.**

Spokane, WA

SS-50 Users: Expand present system to maximum or build from ground up. We provide PCBs for motherboards, interfaces, etc. Write for specs and information. **Quality Research Company, PO Box 7207, Spokane, WA 99207.**

Dealers: Listings are **$15** per month in prepaid quarterly payments, or one yearly payment of **$150**, also prepaid. Ads include 25 words describing your products and services plus your company name, address and phone. (No area codes or merchandise prices, please.) Call Marcia at 603-924-7138 or write *Kilobaud Microcomputing*, Ad Department, Peterborough, NH 03458.

✓See List of Advertisers on page 210

CLASSIFIEDS

Classified advertisements are intended for use by persons desiring to buy, sell or trade used computer equipment. No commercial ads are accepted.

Two sizes of ads are available. The $5 box allows up to 5 lines of about 35 characters per line, including spaces and punctuation. The $10 box allows up to 10 lines. Minimize use of capital letters to save space. No special layouts allowed. Payment is required in advance with ad copy. We cannot bill or accept credit.

Advertising text and payment must reach us 60 days in advance of publication (i.e., copy for March issue, mailed in February, must be here by Jan. 1). The publisher reserves the right to refuse questionable or inapplicable advertisements. Mail copy with payment to: **Classifieds, Kilobaud Microcomputing, Peterborough, NH 03458**. Do not include any other material with your ad as it may be delayed.

For Sale: Heath H-8, serial I/O (cassette and RS-232) and 8K memory boards. Extended BASIC, all documentation. Will throw in as bonus one assembled but untested 12K Econoram VI board. $450 10% down, balance COD/UPS. M. H. Endres, Box 8, Spirit Lake, ID 83869. 208-623-5911.

Must Sell!!: TRS-80, 16K, Level II, tape digitizer, MDX-2 full interface, all game and programming software, programming aids in machine and ASSMBLR, schematic diagrams. Full package $559. Call 303-733-2439 or write to Wendell K. R. Hutchins, 1175 S. Lipan Unit #A, Denver, CO 80223. Shp'd UPS COD.

Needed: Western Electric and Bell System manuals on residential telephones and accessories, including model 500, Trimline, Touch-A-Matic S, speakerfone, jacks, wiring, installation procedures, etc. C. Sarver, 256 West 88th St., New York, NY 10024. 212-874-3529.

For Sale: I have over supply of new diskettes and other computer supplies. Too many to list here. SASE G. R. Davis, 16 W. Walnut Ave., Westmont, NJ 08108.

For Sale: Heath H8 computer 1 year old, config. for O-origin CP/M, H17 dual disk drive, Trionyx 64K mem., and WH8-4 4 port I/O with 3 RS-232 rear panel connectors. Complete documentation and $600 of software. Asking $1990 or best offer, will ship no charge in U.S. Call collect; Dennis Crouse 617-661-8080 or 617-662-7513.

For Sale: Amp IEEE-488 cables @ $25. Axiom EX801 printer (PET)—$250. SYM1 + supply—$160. LP-3 logic probe—$35. 2708's @ $4. 1–24 cursor—$50. J. Brinson, 10136 E. 96th, Indpls, IN 46256.

For Sale: *Microcomputing:* #8 thru current issues, $75 or best offer. *Interface Age:* August 1977 thru current issues, $50 or best offer. Paul Brainerd, 36 Sleigh Rd., Chelmsford, MA 01824. 617-256-6458.

Heath—H8/48K (16 Trionix—32 Heath) 4-port and H8-5 serial with dual Heath cassette. H9 CRT 24 line, upper/lowercase. All for $888. Also WH14 printer $555. Call 201-939-2421. R. Jackson, 201 Mortimer Ave., Rutherford, NJ 07070.

For Sale: computer pc boards for arcade games Pac-man, Space Invaders and Missile Command; pong board with 60+ ICs; unmarked boards with gold ww IC sockets and 2708 EPROMs; 2 miniature arcade games. SASE for more info to Rahm Yazdi, 910 S. Orange Ave., Short Hills, NJ 07078.

32K PET/CBM 2001 with Skyles BASIC utility ROM $600. FOB San Mateo, CA. Elf II, giant with complete software $200. Also SYM-1 with ROM BASIC and assm and case $200. Call collect 415-574-4212, Patrick Young.

For Sale: 1 yr old industrial 19" color monitor. Open frame, 30+ MHz bandwidth. $175. 6502 computer with cage, 3 cards & documents. Color and sound output, $75. Navid Yazdi, 1130 N. 47 Ave. 6, Omaha, NE 68132.

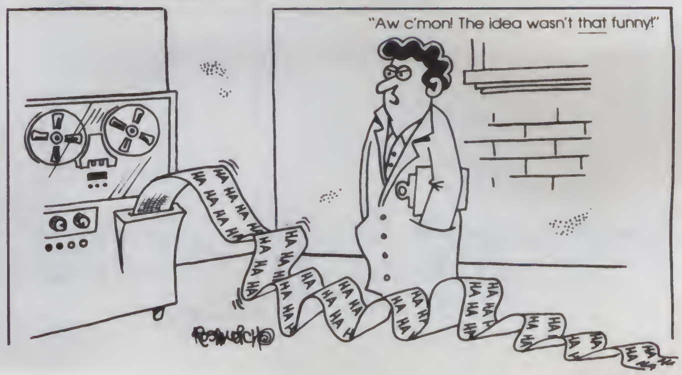
Super magazine sale: *Byte*, May 78, thru May 81, 37 issues; *Microcomputing*, #11 thru #29, #37 thru #54, 37 issues; *Creative Computing*, Jan. 78, thru July 81, 34 issues; *On Computing*, summer 79 thru spring 81, 9 issues; any issue $1.75 each or all of the above for $150. I will pay postage. Call Joe 505-266-2344, or 3503 Berkeley Pl., N.E., Albuquerque, NM 87106.

Wanted: Sorcerer owners for exchange of items. I have 100 programs and much info. Send list of your items. Thanks. Rick Carlsen, 247 Bath Rd.—#710, Kingston, Ont., Canada, K7M 2X9.

For Sale: S-100 boards. Proc Tech VDM-1, $100; 450 ns static mem; 8K Godbout, $80; 4K SSM and Proc Tech, $40; SSM EPROM progr w/EPROMs, $100; modified Hitachi TV monitor, $75; Expandor printer, parallel ASCII, $175. Compl doc. A. Stumpf, 411 E. Palm, Litchfield Pk, AZ 85340. 602-935-2053.

For Sale: Exidy 48K Sorcerer, S-100 expansion unit, North Star dual Micro-Disk system, Leedex monitor and plenty of software. $1450 complete. Will consider separate sales. Digital Graphic Systems CAT-100 video digitizing system. This is a 2 board S-100 device with its own on-board 32K video memory with full documentation and software. $1200. J. Clark, 506 Nadel, Suisun City, CA 94585. 707-426-4582.

"Aw c'mon! The idea wasn't that funny!"



Computer Showcase Expos

A series of regional shows for business, professional and personal users of small computer systems called Computer Showcase Expos will open this autumn. The San Francisco Bay Area Computer Showcase Expo is set for Oct. 21–23 at Brooks Hall. The South Florida Computer Showcase Expo is scheduled for Oct. 30–Nov. 1 at the Miami Expo/Center. The Los Angeles Computer Showcase Expo will be Nov. 13–15 at the Convention Center. For more information, contact the Interface Group, 160 Speen St., Framingham, MA 01701 (617-879-4502; outside Massachusetts, 800-225-4620).

Classroom Applications

A two-day conference on "Classroom Applications of Computers" is set for Oct. 2 and 3 at Independence High School, San Jose, CA. For more information, contact Computer-Using Educators, c/o W. Don McKell, Independence High School, 1776 Educational Park Drive, San Jose, CA 95133 (408-288-7642).

Blacksburg Workshops

A nine-day workshop on electronics is set for late October and early November in Blacksburg, VA. The sessions are Digital Electronics for Automation and Instrumentation, Oct. 26–28; Microcomputer Design Interfacing and Programming Using the Z-80/8085/8080, Oct. 29–31; and Scientific Instrument Automation, Interfacing and Programming Using the TRS-80 Microcomputer, Nov. 2–4. The workshop will be directed by Dr. Paul Field, Dr. Chris Titus, Dr. Jon Titus and David Larsen. For more information, contact Dr. Lindy Leffel, C.E.C., Virginia Tech, Blacksburg, VA 24061 (703-961-5241).

Educational Software Show

Queue's second annual Educational Software Symposium is set for Oct. 17–18 at the Stouffer's Inn, White Plains, NY. The Symposium will feature educational software from dozens of publishers, and a wide variety of seminars, panels, and user interest group meetings on such topics as designing educational software, evaluating educational software, computers in the elementary classroom and computers in various curricula, including math, science, English and foreign language. Speakers will include educators and software authors and publishers. Registration is $45 in advance, $55 at the door. Write to Monica Kantrowitz, President, Queue, Inc., 5 Chapel Hill Drive, Fairfield, CT 06432.

International Electrical, Electronics Conference

The 1981 International Electrical, Electronics Conference and Exposition is scheduled at Exhibition Place in Toronto, Oct. 5, 6 and 7. The show will include over 150 exhibitors, of whom half will be Canadian manufacturers. The keynote speaker will be David A. Golden, Chairman of Telesat Canada. For more information, contact Southex Exhibitions, 1450 Don Mills Road, Don Mills, Ontario M3B 2X7 (416-445-6641).

The Southeast Computer Show

The Southeast Computer Show & Office Equipment Exposition will be Oct. 29–Nov. 1 at the Atlanta Civic Center. The show is produced by National Computer Shows, 824 Boylston St., Chestnut Hill, MA 02167 (617-739-2000).

New Jersey Microcomputer Show

The second annual New Jersey Microcomputer Show and Fleamarket will be on Oct. 24 at the Holiday Inn (north) Convention Center at the Newark International Airport (NJ Turnpike Exit 14). The show is managed by Kengore Corporation, producers of the National TRS-80 Microcomputer Show. For additional information, contact Kengore Corporation, 3001 Route 27, Franklin Park, NJ 08823 (201-297-2526).

Northeast Computer Show

The third annual Northeast Computer Show and Office Equipment Exposition is scheduled for Oct. 15–18 at Boston's Hynes Auditorium. The show is produced by National Computer Shows, 824 Boylston Street, Chestnut Hill, MA 02167 (617-739-2000).

NYSAEDS Conference

On Oct. 18, 19 and 20, the New York State Association for Educational Data Systems (NYSAEDS) will have its annual conference in Syracuse, NY. The theme is Software; the keynote speaker is Marge Kosel from Minnesota Educational Computing Consortium and the banquet speaker is Dr. Earl Joseph (Futurist) from Sperry Rand. For further information, contact Don Ross, Ardsley High School, Ardsley, NY 10502.

Call for Papers

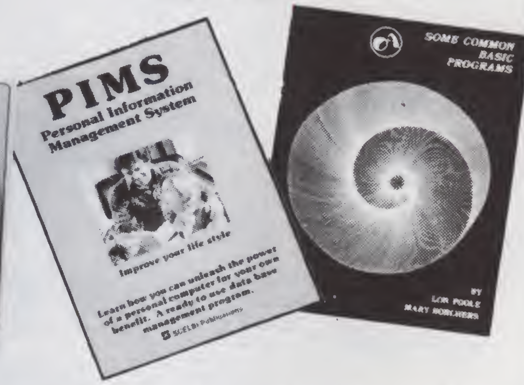
The 1982 SAS Users Group International (SUGI) conference has extended a call for papers for the following sessions: Capacity Planning and Evaluation, and Systems Software (computer capacity management), Evaluation and Consulting (training and consulting in universities, government and private industry), Econometrics and Time Series (modeling, time series analysis and financial analysis and report writing), Graphics (creative uses of graphics in communication and analysis using graphics as a tool), Information Systems (SAS interfaces: databases and database management systems, management information systems, general issues and geographic databases and display), Interactive Techniques (using SAS interactively with TSO and CMS), Statistics (regression: modeling and analysis, nonparametrics and categorical data analysis, analysis of variance applications, multivariate analysis and biometrics and other applied statistical areas) and Posters (graphics and others). SUGI will be Feb. 14–17, 1982, in San Francisco, CA. Deadline for submitting abstracts is October 15, 1981. For more information, contact Helene Cavior, Bureau of Prisons, 330 Primrose Road, Burlingame, CA 94010 (415-347-0721), or SAS Institute, Inc., Box 8000, Cary, NC 27511 (919-467-8000).

programming the Z80
Scelbi "8080" Software Gourmet Guide & Cook Book
6502
Z80
BASIC CODING SHEET
MICROCOMPUTING

kb microcomputing book nook

—BASIC & PASCAL—

—BUSINESS—



new

new

● **INTRODUCTION TO TRS-80 LEVEL II BASIC AND COMPUTER PROGRAMMING**—BK1219—by Michael P. Zabinski. Written by an experienced educator, this is the book for those beginners who want to learn about computers without having to become an expert. It has practical programs, useful line-by-line comments, excellent flowcharts accompanied by line numbers and over 200 exercises which help the reader assess progress, reinforce comprehension, and provide valuable practical experience. $10.95.*

● **50 BASIC EXERCISES**—BK1192—by J. P. Lamoitier. This book is structured around the idea that the best way to learn a language is through actual practice. It contains 50 completely explained exercises: statement and analysis of the problem, flowcharts, programs and actual runs. Program subjects include mathematics, business, games, and operations research, and are presented in varying levels of difficulty. This format enables anyone to learn BASIC rapidly, checking their progress at each step. $12.95*

● **THE BASIC HANDBOOK**—BK1174—by David Lien. This book is unique. It is a virtual ENCYCLOPEDIA of BASIC. While not favoring one computer over another, it explains over 250 BASIC words, how to use them and alternate strategies. If a computer does not possess the capabilities of a needed or specified word, there are often ways to accomplish the same function by using another word or combination of words. That's where the HANDBOOK comes in. It helps you get the most from your computer, be it a "bottom-of-the-line" micro or an oversized monster. $14.95.*

● ***LEARNING LEVEL II***—BK1175—by David Lien. Written especially for the TRS-80, this book concentrates on Level II BASIC, exploring every important BASIC language capability. Updates are included for those who have studied the Level I User's Manual. Sections include: how to use the Editor, dual cassette operation, printers and peripheral devices, and the conversion of Level I programs to Level II. $15.95.*

● **BASIC BASIC (2ND EDITION)**—BK1026—by James S. Coan. This is a textbook which incorporates the learning of computer programming using the BASIC language with the teaching of mathematics. Over 100 sample programs illustrate the techniques of the BASIC language and every section is followed by practical problems. This second edition covers character string handling and the use of data files. $10.50.*

● **ADVANCED BASIC**—BK1000—Applications, including strings and files, coordinate geometry, area, sequences and series, simulation, graphing and games. $10.75*.

● **SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS (2nd Edition)**—BK1073—by Donald Spencer, provides the serious student of BASIC programming with interesting problems and solutions. No knowledge of math above algebra required. Includes a number of game programs, as well as programs for financial interest, conversions and numeric manipulations. $6.95.*

● **PASCAL**—BK1188—by Paul M. Chirlian. Professor Chirlian's textbook combines a simple approach to the PASCAL language with comprehensive coverage on how a computer works, how to use a flowchart, working from a terminal as well as batch operation and debugging. Special attention is paid to idiosyncrasies of the language and syntax flowcharts abound for the convenience of the experienced programmer. Well indexed. $12.95*

● **INTRODUCTION TO PASCAL**—BK1189—by Rodnay Zaks. A step-by-step introduction for anyone wanting to learn the language quickly and completely. Each concept is explained simply and in a logical order. All features of the language are presented in a clear, easy-to-understand format with exercises to test the reader at the end of each chapter. It describes both standard PASCAL and UCSD PASCAL, the most widely used dialect for small computers. No computer or programming experience is necessary. $14.95.*

● **PROGRAMMING IN PASCAL**—BK1140—by Peter Grogono. The computer programming language PASCAL was the first language to embody in a coherent way the concepts of structured programming, which has been defined by Edsger Dijkstra and C.A.R. Hoare. As such, it is a landmark in the development of programming languages. PASCAL was developed by Niklaus Wirth in Zurich; it is derived from the language ALGOL 60 but is more powerful and easier to use. PASCAL is now widely accepted as a useful language that can be efficiently implemented, and as an excellent teaching tool. It does not assume knowledge of any other programming language; it is therefore suitable for an introductory course. $12.95.*

—GAMES—

● **40 COMPUTER GAMES**—BK7381—Forty games in all in nine different categories. Games for large and small systems, and even a section on calculator games. Many versions of BASIC used and a wide variety of systems represented. A must for the serious computer gamesman. $7.95*

● **BASIC COMPUTER GAMES**—BK1074—Okay, so once you get your computer and are running in BASIC, then what? Then you need some programs in BASIC, that's what. This book has 101 games for you from very simple to real buggers. You get the games, a description of the games, the listing to put in your computer and a sample run to show you how they work. Fun. Any one game will be worth more than the price of the book for the fun you and your family will have with it. $7.50.*

● **MORE BASIC COMPUTER GAMES**—BK1182—edited by David H. Ahl. More fun in BASIC! 84 new games from the people who brought you *BASIC Computer Games*. Includes such favorites as Minotaur (battle the mythical beast) and Eliza (unload your troubles on the doctor at bargain rates). Complete with game description, listing and sample run. $7.50.*

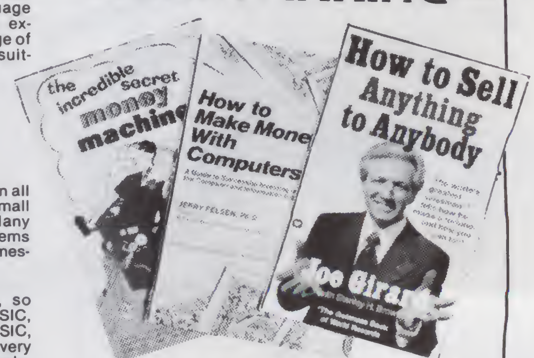
● **WHAT TO DO AFTER YOU HIT RETURN**—BK1071—PCC's first book of computer games . . . 48 different computer games you can play in BASIC . . . programs, descriptions, many illustrations. Lunar Landing, Hammurabi, King, Civel 2, Qubic 5, Taxman, Star Trek, Crash, Market, etc. $10.95.*

● **PAYROLL WITH COST ACCOUNTING—IN BASIC**—BK1001—by L. Poole & M. Borchers, includes program listings with remarks, descriptions, discussions of the principle behind each program, file layouts, and a complete user's manual with step-by-step instructions, flowcharts, and simple reports and CRT displays. Payroll and cost accounting features include separate payrolls for up to 10 companies, time-tested interactive data entry, easy correction of data entry errors, job costing (labor of distribution), check printing with full deduction and pay detail, and 16 different printed reports, including W-2 and 941 (in **CBASIC**). $20.00.*

● **SOME COMMON BASIC PROGRAMS**—BK1053—published by Adam Osborne & Associates, Inc. Perfect for non-technical computerists requiring ready-to-use programs. Business programs, plus miscellaneous programs. Invaluable for the user who is not an experienced programmer. All will operate in the stand alone mode. $14.99 paperback.

● **PIMS: PERSONAL INFORMATION MANAGEMENT SYSTEM**—BK1009—Learn how to unleash the power of a personal computer for your own benefit in this ready-to-use data-base management program. $11.95.*

—MONEYMAKING—



● **HOW TO MAKE MONEY WITH COMPUTERS**—BK1003—In 10 information-packed chapters, Jerry Felsen describes more than 30 computer-related, money-making, high profit, low capital investment opportunities. $15.00.*

● **HOW TO SELL ANYTHING TO ANYBODY**—BK7306—According to *The Guinness Book of World Records*, the author, Joe Girard, is "the world's greatest salesman." This book reveals how he made a fortune—and how you can, too. $2.25.*

● **THE INCREDIBLE SECRET MONEY MACHINE**—BK1178—by Don Lancaster. A different kind of "cookbook" from Don Lancaster. Want to slash taxes? Get free vacations? Win at investments? Make money from something that you *like* to do? You'll find this book essential to give you the key insider details of what is really involved in starting up your own money machine. $5.95.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information.

No C.O.D. orders accepted. All orders add $1.50 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

PRICES SUBJECT TO CHANGE WITHOUT NOTICE

FOR TOLL FREE ORDERING CALL 1-800-258-5473

kb microcomputing book nook

—INTRODUCTORY—

new

●**UNDERSTANDING AND PROGRAMMING MICROCOMPUTERS**—BK7382—A valuable addition to your computing library. This two-part text includes the best articles that have appeared in 73 and Kilobaud Microcomputing magazines on the hardware and software aspects of microcomputing. Well-known authors and well-structured text helps the reader get involved. $10.95*

●**SOME OF THE BEST FROM KILOBAUD MICROCOMPUTING**—BK7311—A collection of the best articles that have appeared in Kilobaud MICROCOMPUTING. Included is material on the TRS-80 and PET systems, CP/M, the 8080/8085/Z-80 chips, the ASR-33 terminal. Data base management, word processing, text editors and file structures are covered too. Programming techniques and hard-core hardware construction projects for modems, high speed cassette interfaces and TVTs are also included in this large format, 200 plus page edition. $10.95*

●**YOUR FIRST COMPUTER**—BK1191—by Rodnay Zaks. Whether you are using a computer, thinking about using one or considering purchasing one, this book is indispensable. It explains what a computer system is, what it can do, how it works and how to select various components and peripheral units. It is written in everyday language and contains invaluable information for the novice and the experienced programmer. (The first edition of this book was published under the title "An Introduction to Personal and Business Computing".) $7.95*

●**MICROPROCESSOR INTERFACING TECHNIQUES**—BK1037—by Austin Lesea & Rodnay Zaks—will teach you how to interconnect a complete system and interface it to all the usual peripherals. It covers hardware and software skills and techniques, including the use and design of model buses such as the IEEE 488 or S-100. $15.95*

●**HOBBY COMPUTERS ARE HERE!**—BK7322—If you want to come up to speed on how computers work... hardware and software... this is an excellent book. It starts with fundamentals and explains the circuits, and the basics of programming, along with a couple of TVT construction projects, ASCII-Baudot, etc. This book has the highest recommendations as a teaching aid. $4.95*

●**THE NEW HOBBY COMPUTERS**—BK7340—This book takes it from where "HOBBY COMPUTERS ARE HERE!" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards... and much, much more! Don't miss this tremendous value! Only $4.95*

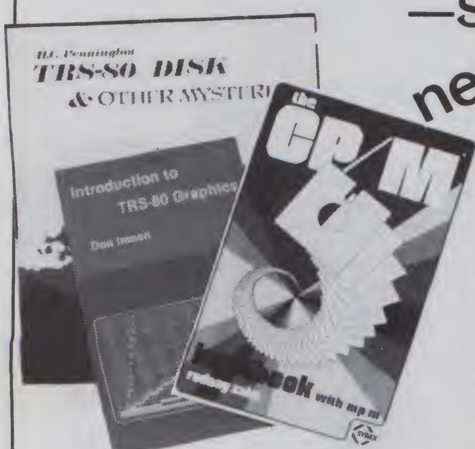
●**AN INTRODUCTION TO MICROCOMPUTERS, VOL. 0**—BK1130—The Beginner's Book—Written for readers who know nothing about computers—for those who have an interest in how to use computers—and for everyone else who must live with computers and should know a little about them. The first in a series of 4 volumes, this book will explain how computers work and what they can do. Computers have become an integral part of life and society. During any given day you are affected by computers, so start learning more about them with Volume 0. $7.95*

●**VOL. I**—BK1030—**2nd Edition completely revised.** Dedicated to the basic concepts of microcomputers and hardware theory. The purpose of Volume I is to give you a thorough understanding of what microcomputers are. From basic concepts (which are covered in detail), Volume I builds the necessary components of a microcomputer system. This book highlights the difference between minicomputers and microcomputers. $12.99*

●**VOL. II**—BK1040 (with binder)—Contains descriptions of individual microprocessors and support devices used only with the parent microprocessor. Volume II describes all available chips. $31.99*

●**VOL. III**—BK1133 (with binder)—Contains descriptions of all support devices that can be used with any microprocessor. $21.99*

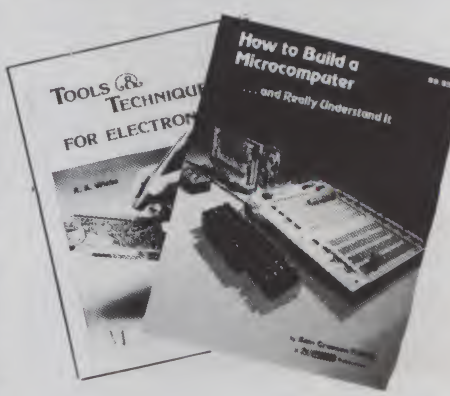
—SPECIAL INTERESTS—



new

●**THE CUSTOM TRS-80 AND OTHER MYSTERIES**—BK1218—by Dennis Kitsz. More than 300 pages of TRS-80 customizing information. With this book you'll be able to explore your computer like never before. Want to turn an 8 track into a mass storage unit? Individual reverse characters? Replace the BASIC ROMs? Make Music? High speed, reverse video, Level I and Level II? Fix it if it breaks down? All this and much, much more. Even if you have never used a soldering iron or read a circuit diagram, this book will teach you how! This is the definitive guide to customizing your 80! $29.95*

●**BASIC FASTER AND BETTER AND OTHER MYSTERIES**—BK1221—by Lewis Rosenfelder. You don't have to learn assembly language to make your programs run fast. With the dozens of programming tricks and techniques in this book you can sort at high speed, swap screens in the twinkling of an eye, write INKEY routines that people think are in assembly language and add your own commands to BASIC. Find out how to write elegant code that makes your BASIC really hum, and explore the power of USR calls. $22.50* *Available end of August.*



●**TRS-80 DISK AND OTHER MYSTERIES**—BK1181—by Harvard C. Pennington. This is the definitive work on the TRS-80 disk system. It is full of detailed "How to" information with examples, samples and in-depth explanations suitable for beginners and professionals alike. The recovery of one lost file is worth the price alone. $22.50*

●**MICROSOFT BASIC DECODED AND OTHER MYSTERIES**—BK1186—by James Farvour. From the company that brought you *TRS-80 DISK AND OTHER MYSTERIES*! Contains more than 6500 lines of comments for the disassembled Level II ROMs, six additional chapters describing every BASIC subroutine, with assembly language routines showing how to use them. Flowcharts for all major routines give the reader a real insight into how the interpreter works. $29.50.

●**THE CP/M HANDBOOK (with MP/M)**—BK1187—by Rodnay Zaks. A complete guide and reference handbook for CP/M—the industry standard in operating systems. Step-by-step instruction for everything from turning on the system and inserting the diskette to correct user discipline and remedial action for problem situations. This also includes a complete discussion of all versions of CP/M up to and including 2.2, MP/M and CDOS. $14.95*.

●**INTRODUCTION TO TRS-80 GRAPHICS**—BK1180—by Don Inman. Dissatisfied with your Level I or Level II manual's coverage of graphics capabilities? This well-structured book (suitable for classroom use) is ideal for those who want to use all the graphics capabilities built into the TRS-80. A tutorial method is used with many demonstrations. It is based on the Level I, but all material is suitable for Level II use. $8.95*

●**TOOLS & TECHNIQUES FOR ELECTRONICS**—BK7348—by A. A. Wicks is an easy-to-understand book written for the beginning kit builder as well as the experienced hobbyist. It has numerous pictures and descriptions of the safe and correct ways to use basic and specialized tools for electronic projects as well as specialized metal working tools and the chemical aids which are used in repair shops. $4.95*

●**HOW TO BUILD A MICROCOMPUTER—AND REALLY UNDERSTAND IT**—BK7325—by Sam Creason. The electronics hobbyist who wants to build his own microcomputer system now has a practical "How-To" guidebook. This book is a combination technical manual and programming guide that takes the hobbyist step-by-step through the design, construction, testing and debugging of a complete microcomputer system. Must reading for anyone desiring a true understanding of small computer systems. $9.95*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information.

No C.O.D. orders accepted. All orders add $1.50 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

PRICES SUBJECT TO CHANGE WITHOUT NOTICE

FOR TOLL FREE ORDERING CALL 1-800-258-5473

AIM 65 BY ROCKWELL INTERNATIONAL

M 65 is fully assembled, tested and warranted. With the
ldition of a low cost, readily available power supply, it's
ady to start working for you.
AIM 65 features on-board thermal printer and
phanumeric display, and a terminal-style keyboard. It
s an addressing capability up to 65K bytes, and comes
th a user-dedicated 1K or 4K RAM. Two installed 4K
DMS hold a powerful Advanced Interface Monitor
ogram, and three spare sockets are included to expand
-board ROM or PROM up to 20K bytes.
An Application Connector provides for attaching a TTY
d one or two audio cassette recorders, and gives exter-
l access to the user-dedicated general purpose I/O lines.
Also included as standard are a comprehensive AIM 65
er's Manual, a handy pocket reference card, an R6500
rdware Manual, an R6500 Programming Manual and an
M 65 schematic.
AIM 65 is packaged on two compact modules. The
cuit module is 12 inches wide and 10 inches long, the
yboard module is 12 inches wide and 4 inches long.
ey are connected by a detachable cable.

ERMAL PRINTER
ost desired feature on low-cost microcomputer systems . . .
Wide 20-column printout
Versatile 5 x 7 dot matrix format
Complete 64-character ASCII alphanumeric format
Fast 120 lines per minute
Quite thermal operation
Proven reliability

LL-SIZE ALPHANUMERIC KEYBOARD
ovides compatibility with system terminals . . .
Standard 54 key, terminal-style layout
26 alphabetic characters
10 numeric characters
22 special characters
9 control functions
3 user-defined functions

UE ALPHANUMERIC DISPLAY
ovides legible and lengthy display . . .
20 characters wide
16-segment characters
High contrast monolithic characters
Complete 64-character ASCII alphanumeric format

PROVEN R6500 MICROCOMPUTER SYSTEM DEVICES
Reliable, high performance NMOS technology . . .
- R6502 Central Processing Unit (CPU), operating at 1 MHz. Has 65K address capability, 13 addressing modes and true index capability. Simple but powerful 56 instructions.
- Read/Write Memory, using R2114 Static RAM devices. Available in 1K byte and 4K byte versions.
- 8K Monitor Program Memory, using R2332 Static ROM devices. Has sockets to accept additional 2332 ROM or 2532 PROM devices, to expand on-board Program memory up to 20K bytes.
- R6532 RAM-Input/Output-Timer (RIOT) combination device. Multipurpose circuit for AIM 65 Monitor functions.
- Two R6522 Versatile Interface Adapter (VIA) devices, which support AIM 65 and user functions. Each VIA has two parallel and one serial 8-bit, bidirectional I/O ports, two 2-bit peripheral handshake control lines and two fully-programmable 16-bit interval timer/event counters.

BUILT-IN EXPANSION CAPABILITY
- 44-Pin Application Connector for peripheral add-ons
- 44-Pin Expansion Connector has full system bus
- Both connectors are KIM-1 compatible

TTY AND AUDIO CASSETTE INTERFACES
Standard interface to low-cost peripherals . . .
- 20 ma. current loop TTY interface
- Interface for two audio cassette recorders
- Two audio cassette formats: ASCII KIM-1 compatible and binary, blocked file assembler compatible

ROM RESIDENT ADVANCED INTERACTIVE MONITOR
Advanced features found only on larger systems . . .
- Monitor-generated prompts
- Single keystroke commands
- Address independent data entry
- Debug aids
- Error messages
- Option and user interface linkage

ADVANCED INTERACTIVE MONITOR COMMANDS
- Major Function Entry
- Instruction Entry and Disassembly
- Display/Alter Registers and Memory
- Manipulate Breakpoints
- Control Instruction/Trace
- Control Peripheral Devices
- Call User-Defined Functions
- Comprehensive Text Editor

LOW COST PLUG-IN ROM OPTIONS
- 4K Assembler—symbolic, two-pass , A65-010 $79.00
- 8K BASIC Interpreter A65-020 $99.00

POWER SUPPLY SPECIFICATIONS
- +5 VDC ± 5% regulated @ 2.0 amps (max)
- +24 VDC ± 15% unregulated @ 2.5 amps (peak) 0.5 amps average

PRICE: $389.00 (1K RAM) **$439.00 (4K RAM)**
Plus $4.00 UPS (shipped in U.S. must give ***street*** address), $10 parcel post to APO's, FPO's, Alaska, Hawaii, Canada, $25 air mail to all other countries
AIM 65 USER MANUAL $5.00 plus $1.50 shipping & handling.
We manufacture a complete line of high quality expansion boards. Use reader service card to be added to our mailing list, or U.S. residents send $1.00 (International send $3.00 U.S.) for airmail delivery of our complete catalog.

We've Expanded Again!
Note our new address.

RNB ENTERPRISES INCORPORATED ✓52

4030 N. 27th Avenue, Suite D
Phoenix, Az 85017
(602) 265-7564

master charge VISA

PRINTERS

10 DAY FREE RETURN

MONITORS TERMINALS

CONFIGURATION HELP

CALCULATORS MISC

TOLL FREE ORDERING

SYSTEMS SOFTWARE

LEASING AVAILABLE

EPSON MX-80
Now in stock!
The MX-80 dot matrix printer. Unequalled Epson reliability. Has all the features of the MX-70 plus more power and extra functions.

C-ITOH STARWRITER: LETTER QUALITY PRINTING FOR UNDER $2000!/This daisywheel printer gives high quality at a low price. 25 cps. Parallel and serial interfaces available.

NEC SPINWRITER FROM THE FIRST NAME IN LETTER QUALITY PRINTERS/Compumart offers beautiful print quality with NEC Spinwriter Terminals. We carry all models from RO THRU KSR WITH NUMERIC KEYPAD—5510-5530. All versions give unsurpassed hard copy output!

NEW INTEGRAL DATA'S 560 PRINTER/All the exciting features of the 400 series plus 14 1/2" paper capacity. 132 col. graphics printer.
IDS 445. Priced lower than the 440 and equipped with a better print head. With & w/o graphics.
IDS 460. Features include correspondence quality printing, high resolution graphics.

Spinwriter 5520

CENTRONICS PRINTERS
3 SERIAL MODEL 737
The closest thing to letter quality print for under $1000.
List $1045 **SPECIAL $795.**
737-1 Parallel Interface
List $995 **SALE PRICE $695.**
PLUS EXCITING REBATE OFFER ON CENTRONICS PRINTERS.

OMNI 810 PRINTER FROM TEXAS INSTRUMENTS **CALL US**
The 820 RO PACKAGE includes machine mounted paper tray and cable. A compressed print option and device forms control are standard features.
THE 820 KSR PACKAGE includes fully ASCII Keyboard plus all of the features of the RO.

CLEARANCE ZENITH COLOR VIDEO MONITOR **$349.**

SUPER SELLING TERMINALS FROM LEAR SIGLER/We have the following Lear Siegler terminals in stock at prices too low to print! Call for quotes.

ADM-3A/Industry's favorite dumb terminal for some very smart reasons/
ADM-3A + NEW from Lear Siegler. CALL!
IT IS HERE! It is the new Intermediate Terminal from Lear Siegler.

SANYO MONITORS AT LOW COMPUMART PRICES/Sanyo's new line of CRT data display monitors are designed for the display of alphanumeric or graphic data.
9" SANYO B/W **$169.**
12" SANYO B/W
12" SANYO W/ GREEN SCREEN
13" SANYO COLOR

Sanyo 13" Color Monitor

NEC COLOR MONITOR/RECEIVER HIGH RESOLUTION/
Composite video using BNC connectors, 8-Pin connector for VCR/VTR video loop In/Out and television reception.

Visit our giant
ANN ARBOR STORE
1250 North Main Street
Ann Arbor, Michigan

FREE CATALOGS

MICRO
The original and most complete catalog of micro-computers, accessories and peripherals.

DEC PDP/LSI-11 Systems configured and integrated with other manufacturers compatibles. The first, best DEC based systems catalog.

Send for them!

HP-41C CALCULATORS
MEMORY MODULES for storing programs of up to 2000 lines of program memory.
"EXTRA SMART" CARD READER. Records programs and data back onto blank magcards.
THE PRINTER. Upper and lower case. High resolution plotting. Portable thermal operation.
APPLICATION MODULES

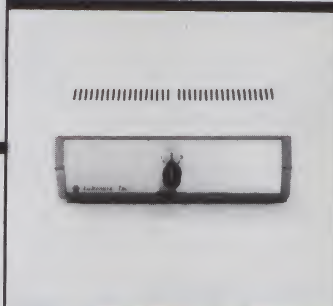
NEW SUPER 41-CV SYSTEMS with Quad RAMS built-in. Maximum memory on-board leaves slots open for Application Pacs and peripherals.
+ CARD READER
+ CARD READER + PRINTER
QUAD RAMS equivalent to four Memory Modules all packed into one.

MATROX PRODUCTS/Compumart stocks the complete line.

DEC LSI-11/Compumart now offers the entire product line.
CALL FOR PRICES AND DELIVERY

NOVATION CAT ACOUSTIC MODEM Answer Originate

NEW! D-CAT Direct Connect Modem from Novation.



Giltronix Switch

NEW!
GILTRONIX RS 232 SWITCH/
The ultimate in flexibility. You can connect three peripherals to one computer or three computers to one peripheral. Switches the eight most important RS 232 signals.

DYSAN DISKETTES/Single side, single density. Hard or Soft Sector **$5. ea.**

MEMOREX 3401's/5 1/4" disks **$3.25.**
/with hub ring for Apple **$3.50.**

MEMORY INTEGRATED CIRCUITS/ Call for quantity discounts when ordering over 50 units.

MOTOROLA 4116 (200 Nano-second Plastic) **$4.50**

RM EXPANSION ACCESSORIES FOR AIM—
CALL SPECS AND PRICES

APPLE III IS IN STOCK/Apple III Information Analyst Package—128K Apple III, Black and White Monitor 12", and information analyst software.

TOP SOFTWARE PACKAGES FROM COMPUMART

VISICALC/FOR APPLE/FOR HP/FOR COMMODORE/FOR ATARI

SOFTWARE FROM APPLE/Apple Plot (the perfect graphic complement for Visicalc/Dow Jones News & Quotes/Apple Fortrom/Apple Writer/Pascal Language System/ Controller Business System

PERSONAL SOFTWARE/Visidex/ VisiTrend/VisiPlot/VisiTerm
MUSE/Super Text

Apple Software

MOUNTAIN COMPUTER/

Expansion accessories for Apple/ Super Talker/The Music System/ ROM plus board with Keyboard filter/ROM Writer/Clock Calendar/AtoD and D to A Converter/ Clock for Apple/CPS Multifunction Board

VIDEX/Video Term (80 col. x 24 line, 7x9 Matrix plug in compatible board for Apple II) w/wo graphics EPROM/SSM Serial & Parallel, Apple Interface/ABT's Numeric Key Plan/California Microcomputer Keyboard

NEW!

VIC 20 PERSONAL COMPUTER FROM COMMODORE
$299.

SEE US AT THE NORTHEAST COMPUTER SHOW, OCTOBER 15-18, 1981, BOOTH #608

IMPORTANT ORDERING INFORMATION
CALL: 800 343-5504, in Massachusetts (617) 491-2700, phones open from 8:30 a.m. to 7:00 p.m. Mon-Fri, 11:00 a.m. to 4:00 p.m. Sat.
PO's: Accepted from Dun & Bradstreet rated companies—shipment contingent upon receipts of signed purchase order.
SALE PRICES: Valid for month of magazine date only—all prices subject to change without notice. Our Ann Arbor retail store is open 11:00 a.m. to 7:00 p.m. Tues-Fri, 10:00 a.m. to 5:00 p.m. on Saturdays.

SYSTEMS APPLE II STARTER

$1895

SAVE OVER $200 ON OUR BEST SELLING APPLE SYSTEM/ System includes a 48K Apple II, Apple Disk, DOS 3.3 and Controller and a Sup R MOD RF Modulator.
List $2209.

EXCLUSIVE FOR THE APPLE: Magic Wand, Videx, Z-80 softcard (Requires 48K Apple and disk).
COMPLETE SUB-SYSTEM $925.

APPLE ACCESSORIES

CHOOSE FROM: Silentype Printer w/x face/Light Pen/Easy Writer (80 col. need a Videx)/Clock for Apple.
FROM MICROSOFT: 16K RAM Board/FORTRAN.
FROM COMPUTER STATION: Hi-Res Dump for 460 Printer.

OUR APPLE INVENTORY IS COMPLETE. WE'VE GOT IT ALL—

apple computer
Authorized Dealer
Accept No Less

SYSTEMS COMMODORE WORD CRAFT 80

$4695

CRISP LETTER QUALITY OUTPUT UNSURPASSED EASE OF OPERATION
This Compumart/Commodore system includes a **COMMODORE 8032 32K CPU**, a **4040 DUAL DISK**, a **C-ITOH PRINTER** and x/face and **WORD PRO 4 PLUS** (all cables included).
List $5685.

EDUCATORS : COMMODORE HAS EXTENDED ITS 3 FOR 2 DEAL

A COMPLETE SYSTEM: includes a **DUAL DISK DRIVE/TRACTOR PRINTER** and an **80 COLUMN 32K CPU.** No interfaces needed. Cables included.
List $3985.
COMPUMART $3635.

SAVE $200 ON COMMODORE ACCESSORIES WITH PURCHASE OF A 32K PET—SAVE $100 WITH A 16K PET.

CHOOSE FROM:
Visicalc/Word Pro 4/Wordcraft 80/ Ozz the Information Wizzard/ Dow Jones Portfolio Mgmt System/ Assembler Development Package.

commodore
Authorized Dealer
Accept No Less

SYSTEMS HP 85 BASIC

$3250

This Basic 16k BYTE system (expandable to 32k) includes: **CRT DISPLAY, THERMAL PRINTER, MAGNETIC TAPE CARTRIDGE, 6 ROM CAPACITY VIA ROM DRAWER, 4 IO PORTS. COMMUNICATION—HPIB, RS232**

PLUS

YOUR CHOICE OF 4 SOFTWARE PACS OR TWO OF OUR "BEST SELLERS"—(VISICALC, GRAPHICS PRESENTATION, INFORMATION MANAGEMENT)
SAVE $400

HP-83 SYSTEM
This Basic System from Hewlett-Packard includes **HP-83/ROM DRAWER/MASS STORAGE ROM/ 51/4" SINGLE MASTER FLEXIBLE DISK DRIVE/HP-1B INTERFACE MODULE/2 METER HP-1B CABLE.**
COMPLETE SYSTEM $4249.

HEWLETT PACKARD
Authorized Dealer
Accept No Less

SYSTEMS AIM 65 STARTER

$795

FROM ROCKWELL
Our **AIM** Starter System for Educational & Laboratory use includes **4K AIM/BASIC & ROM/ASSEMBLER & ROM/POWER SUPPLY EG1 ENCLOSURE/CRAIG TAPE RECORDER.**

ACCESSORIES FOR AIM STARTER: PL 65 High Level Language/Paper for the Aim (roll)/Rockwell's 4 slot Motherboard/

WE ALSO CARRY RM EXPANSION ACCESSORIES FOR THE AIM-65.

Rockwell International
Authorized Dealer
Accept No Less

VISA
MasterCard

Systems Customized

800-343-5504

IN MASS CALL 617-491-2700

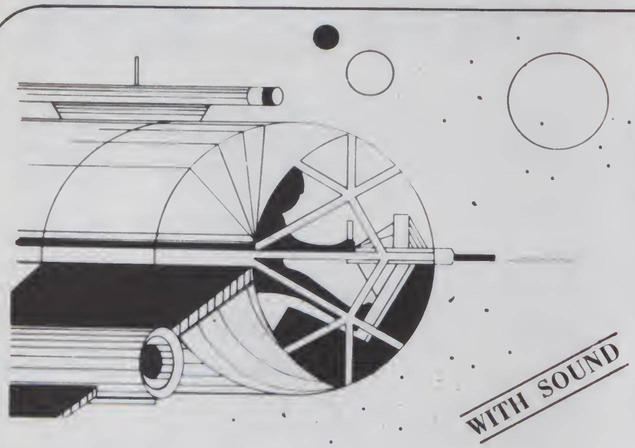
COMPUMART 43

65 Bent Street, Dept 129
PO Box 568, Cambridge, MA 02139

From THE LEADER...
We just might be the largest independent small systems dealer in the country. Here's why:
COMPUMART has been serving the computer needs of industry since 1971.
We stock, for immediate shipment, only those products from the finest micro-computer manufacturers.
And any product, except software, can be returned within 10 days for a full refund—even if you just change your mind. We also honor all manufacturers' warranties. Our expert technicians will service any product we sell
Call us for more information on products, product configuration and service. Our phones are open Monday thru Friday, 8:30 a.m. to 7:00 p.m. and Saturday 11:00 a.m. to 4:00 p.m.
We have a staff of highly knowledgeable sales people waiting to hear from you, and to help. Because service is what we're all about.

GO BOLDLY . . .

Where No TRS-80* Program Has Gone Before!



DANGER IN ORBIT

DATE: 28.02.2047
LOCATION: 270 million miles from Terra
MISSION: Maintaining Terra's Space Lanes

Briefing will follow;
1.1 Your mission is to destroy any asteroids in your sector and to prevent alien spacecraft from infiltrating the Terran Defense Network.
1.2 Your ship is armed with an anti-matter cannon. You can shoot large asteroids, but this turns them into many smaller asteroids, each capable of destroying your ship.
1.3 In addition, alien ships can make instantaneous hyperspace jumps into your area and start firing on your ship.
1.4 You'll need lightning reflexes and nerves of steel to survive Danger In Orbit. We have no use for non-survivors!

Danger In Orbit, a real-time, machine-language game, features variable levels of difficulty, superb high-speed graphics, sound effects and automatic score keeping. (T1 or T2)

Order No. 0237R
$14.95 Tape.**
Order No. 0247RD
$19.95 Disk.**

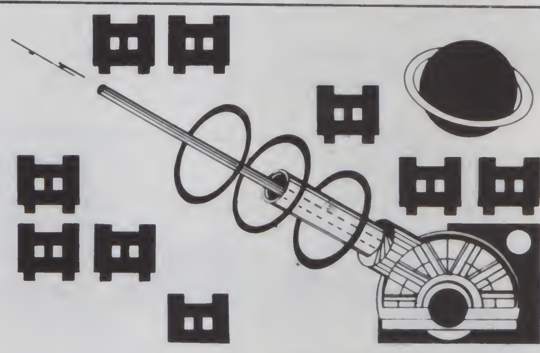
BALL TURRET GUNNER

For years the Petro Resource Conglomerate has attacked our photon collection stations and strangled our deep-space trade routes. The PRC Exxonerator Class light fighters (code name: Gnat) have been their main weapon. Now you can strike back, by joining the Ball Turret Gunner Service.

Imagine yourself at the control console of an LW-1417 Stratoblazer (Type B Strategic Laser Weapon). Your Hindsight Director informs you that a Gnat fighter is coming in for an attack. You pivot your gigawatt laser turret until you can see the target on your monitor. The Range Indicator shows him coming in fast. The Targeting Computer studies his course and speed as your finger tenses over the firing key. You know you'll have only a fraction of a second in which to react. The Gnat fighter's evasive maneuvers cause him to dance in your sights. Suddenly, you see the FIRE Command and you react instinctively. Your laser beam lashes out and reduces the Gnat to an expanding ball of ionized gas. Mission accomplished!

Ball Turret Gunner, with your choice of multiple levels of difficulty, optional sound effects and superb graphics, is more than just a game. It's an adventure. Experience it! (T1)

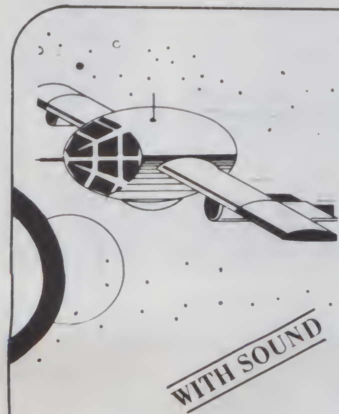
Order No. 0051R $9.95.**



ALIEN ATTACK FORCE

The invaders are coming! Earth's defenses are dead *except for your* Laser base. Your assignment is to destroy the approaching invaders before they destroy Earth. Before Earth's sensors failed, they detected 550 armed invaders in space, speeding toward us in 10 attack formations of 55 in each group. The sensors detected four different types of attack craft: Large, Medium, Small, and short profile craft which is the most difficult to destroy. If you cannot stop these space attackers they will stop Earth . . . for good. (T1)

Order No. 0240R $9.95.**



COSMIC PATROL

WARNING: PLAYERS OF THIS GAME SHOULD BE PREPARED FOR A STATE OF REALISM HITHERTO UNAVAILABLE ON THE TRS-80

Skilled players soon master many difficult computer games, but COSMIC PATROL is in a world all its own. The challenge intensifies! Supporting graphics and sound (optional) make each encounter an exciting new experience. It all adds up to a Super 3-S package. . . skill, sight and sound.

Scenario: The Cosmic Patrol program puts you in the command chair of a small interstellar patrol craft. Your mission is to defend Terran space and prey on the Quelon supply ships which carry essential parts and lubricants for that implacably hostile robotic force. The drone freighters are fairly easy pickings for the accomplished starship pilot, but beware of the I-Fighter escorts. They're armed, fast and piloted by intelligent robots linked to battle computers. They *never* miss.

The Cosmic Patrol program is not just another search and destroy game. With its fast, real-time action, impressive sound option and superb graphics, this machine-language program is the best of its genre.

Don't keep putting quarter after quarter into arcade games or spending big bucks for video game cartridges. Get Cosmic Patrol from Instant Software— and get the best for less! (T1 or T2)

Order No. 0223R
$14.95 Tape.**
Order No. 0224RD
$19.95 Disk.**

✓405

(T1) = TRS-80 Model I, Level II, 16K RAM.

(T2) = TRS-80 Model I, Level II 16K, expansion interface 16K + 1 disk drive.

TO ORDER

See Your Local Instant Software Dealer or Call Toll-Free 1-800-258-5473
(Orders Only)
In NH dial 1-603-924-7296

Instant Software™

PETERBOROUGH, N.H. 03458

*TRS-80 is a registered trademark of the Tandy Corporation
**Plus shipping

PROVEN Computing SOLUTIONS

We give you more of what you buy a computer for: to gain a competitive edge in your industrial, commercial, or scientific application. While other computers try to make the best of slow memory, slow processors, and me-too engineering, **CompuPro** is delivering solutions for today's commercial computing ...proven solutions that create expandable, modular, fast, exceptionally reliable, and truly professional level machines that conform fully to the IEEE 696/S-100 standards. We back all products with a minimum 1 year limited warranty.

When you depend on your computer, choose a computer on which you can depend: S-100 from CompuPro.

1 **DISK 1, A SUPERB DISK CONTROLLER.** Disk controllers don't have to be your system's bottleneck: **Disk 1** is lightning fast, thanks to properly implemented DMA (with arbitration) and data transfer that's independent of CPU speed. Handles up to four 8" or 5.25" floppy disk drives, single or double-sided, single or double density (soft sectored). With BIOS for CP/M-80*. **Manual available for $25. A/T $495, CSC $595. Coming soon: Disk 2, the fast DMA hard disk controller.**

2 **CP/M-86.** Available only with **Disk 1** purchase. Supplied on diskette with full documentation; ready for load and go operation. **$300.**

3 **OASIS* OPERATING SYSTEM.** Single user (**$500**) and multi-user (**$850**) available for use with **CPU Z.** Supplied on diskette with full documentation. **Coming soon: OASIS 16** for use with 16 bit systems.

4 **COMPLETE DUAL DISK DRIVE SYSTEM.** With one **Disk 1** board, desktop dual drive enclosure with Shugart SA-800 series drives, and power supply cable. Also includes CP/M-80. **Introductory special: $2195.**

5 **SYSTEM SUPPORT 1.** Includes battery operated clock/calendar; sockets for battery backup RAM, optional 9511 or 9512 math processor, and 2716 ROM; RS-232 serial port; triple interval timers; dual interrupt controllers; power fail interrupt; and more. Comprehensive owner's manual includes numerous software examples. **$295 Unkit, $395 A/T, $495 CSC** (add $195 to the above prices for the optional math processor).

6 **8 BIT CPU Z.** A Z80*-based CPU board that includes all standard features plus many convenience options. Meets all IEEE 696/S-100 specifications, including timing. **$225 Unkit, $295 A/T** (both operate at 4 MHz), **$395 CSC** (with 6 MHz CPU).

7 **16/8 BIT CPU 8085/88.** Now there's an easy way to gain 16 bit capability without sacrificing your library of 8 bit software: **CPU 8085/88** combines an 8088 CPU to handle 16 bit software and an 8085 CPU for 8 bit software. **$295 Unkit, $425 A/T** (both operate at 5 MHz); **$525 CSC** (with 6 MHz 8085 and 8088).

8 **8088/8086 MONITOR-DEBUGGER SOFTWARE.** Ideal for use with **CPU 8085/88.** Supplied on single sided, single density, soft sectored 8" disk. CP/M-80 compatible (not required if you have CPM/86). Great development tool; mnemonics used in debug conform as closely as possible to current CP/M DDT mnemonics. **$35.**

9 **8 BIT CPU 8085.** Same as **CPU 8085/88,** but without 8088. May be easily upgraded to 16 bit operation. **$235 Unkit, $325 A/T, $425 CSC.**

10 **INTERFACER 1.** Two RS-232 serial ports, with full handshaking and independently selectable Baud rates, connect your computer to serial peripherals. **$199 Unkit, $249 A/T, $324 CSC.**

11 **INTERFACER 2.** Adds three full duplex parallel ports, and one serial port with all the features of an **Interfacer 1** serial port, to your computer. **$199 Unkit, $249 A/T, $324 CSC.**

12 **INTERFACER 3-5.** Includes five RS-232 serial ports (2 synchronous/asynchronous, 3 asynchronous) with full handshaking. Includes software programmable Baud rates and many other convenience features. **$599 A/T, $699 CSC.**

13 **INTERFACER 3-8.** Same as above, but with eight full RS-232 serial ports (2 synchronous/asynchronous, 6 asynchronous). Ideal for multi-user/multi-terminal systems. **$699 A/T, $849 CSC.**

14 **SPECTRUM COLOR GRAPHICS BOARD.** With 8K of IEEE-compatible static RAM, full duplex bidirectional I/O port for keyboard or joystick interface, and 6847-based graphics generator that can display all 64 ASCII characters. 10 modes of operation, from alphanumerics/semi-graphics in 8 colors to dense 256 X 192 full graphics. **$299 Unkit, $399 A/T, $449 CSC.**

15 **6 SLOT HIGH SPEED MOTHERBOARD.** With all edge connectors, Faraday shielding, extensive bypassing, and split active termination. **$89 Unkit, $129 A/T.**

16 **12 SLOT HIGH SPEED MOTHERBOARD.** Same as above, but with 12 slots. **$129 Unkit, $169 A/T.**

17 **20 SLOT HIGH SPEED MOTHERBOARD.** Same as above, but with 20 slots. **$174 Unkit, $214 A/T.**

18 **MEMORY MANAGER.** Extends the addressing of older S-100 machines beyond 64 KBytes to allow for greatly expanded memory capacity. **$59 Unkit, $85 A/T, $100 CSC.**

19 **ACTIVE TERMINATOR.** Promotes reliable and accurate data transfer in older machines by properly terminating bus lines, thereby minimizing ringing, crosstalk, overshoot, noise, and other gremlins associated with unterminated lines. **$34.50 Unkit, $59.50 A/T.**

20 **2708 EPROM BOARD.** This board is **the** way to store often used routines or pieces of software. **$85 Unkit, $135 A/T, $195 CSC** (2708s not included).

21 **DOCUMENTATION.** User manuals are available for all products. **Interfacer 3, $10; Disk Controller, $25; System Support 1, $20;** most others, **$5.** Also available: **"CompuPro Product User Manuals: 1975 - 1980".** This 250+ page book includes data on all older **Godbout/CompuPro** products, as well as many newer products such as the **Spectrum Color Graphics** board, **Interfacers 1** and **2, CPU 8085/88,** motherboards, **CPU Z,** and more. Whether you want to evaluate the innovative engineering behind **CompuPro** products, or study the nuts and bolts of high speed computer operation, this is the book for you. **$20** plus shipping.

22 **RAM 16-64.** Configurable as 64K X 8 for 8 bit systems or 32K X 16 for 16 bit systems. Static, IEEE-696/S-100 compatible, extended addressing, 10 MHz operation. **$1195 A/T, $1295 CSC.**

23 **RAM 17-64.** The lowest power (¼ Amp typical) 64K X 8 static board in the business. IEEE-696/S-100 compatible, 4 optional 2K windows, no wait states with 6 MHz Z80. **$1095 A/T, $1195 CSC, $875 Unkit.**

24 **RAM 20-32.** For lower density applications. Static, IEEE-696/S-100 compatible, extended addressing or bank select, up to 10 MHz operation. **$559 A/T, $659 CSC, $449 Unkit.** 8K and 16K versions also available.

Most **CompuPro** products are available in **Unkit** form, **Assembled/Tested** (both with 1 year warranty), or qualified under the high-reliability **Certified System Component** (CSC) program (200 hour burn-in, 2 year extended warranty, more). Note: Unkits are not intended for novices, as de-bugging may be required due to problems such as IC infant mortality. Factory service is available for **Unkits** at a flat service charge.

*LEGAL CORNER: **Z80** is a registered trademark of Zilog; **CP/M** is a registered trademark of Digital Research; **OASIS** is a trademark of Phase 1.

How to Order: Call **415-562-0636** for the name of the authorized **CompuPro** sales center nearest you, or for placing factory direct VISA®/Mastercard® orders. Prices shown do not include tax, shipping charges, or dealer installation/support services.

CompuPro™

✓42

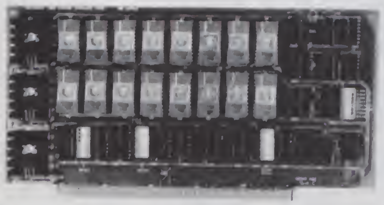
division

GODBOUT ELECTRONICS

OAKLAND AIRPORT, CA 94614 (415) 562-0636

DIGITAL RESEARCH COMPUTERS
(214) 271-3538

32K S-100 EPROM CARD
NEW!



$79.95
KIT

USES 2716's

Blank PC Board - $34

ASSEMBLED & TESTED
ADD $30

SPECIAL: 2716 EPROM's (450 NS) Are $9.95 Ea. With Above Kit.

KIT FEATURES:
1. Uses +5V only 2716 (2Kx8) EPROM's.
2. Allows up to 32K of software on line!
3. IEEE S-100 Compatible.
4. Addressable as two independent 16K blocks.
5. Cromemco extended or Northstar bank select.
6. On board wait state circuitry if needed.
7. Any or all EPROM locations can be disabled.
8. Double sided PC board, solder-masked, silk-screened.
9. Gold plated contact fingers.
10. Unselected EPROM's automatically powered down for low power.
11. Fully buffered and bypassed.
12. Easy and quick to assemble.

32K SS-50 RAM

$299.00 KIT

For 2MHZ
Add $10

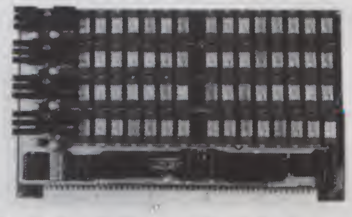
Blank PC Board
$50

For SWTPC
6800 - 6809 Buss

Support IC's
and Caps
$19.95

Complete Socket Set
$21.00

Fully Assembled,
Tested, Burned In
Add $30



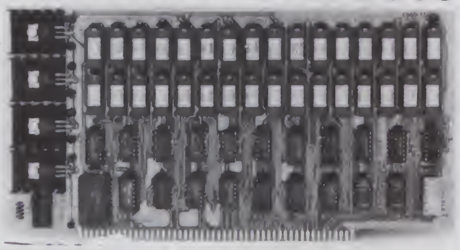
At Last! An affordable 32K Static RAM with full 6809 Capability.

FEATURES:
1. Uses proven low power 2114 Static RAMS.
2. Supports SS50C - EXTENDED ADDRESSING.
3. All parts and sockets included.
4. Dip Switch address select as a 32K block.
5. Extended addressing can be disabled.
6. Works with all existing 6800 SS50 systems.
7. Fully bypassed. PC Board is double sided, plated thru, with silk screen.

16K STATIC RAM KIT-S 100 BUSS

PRICE CUT!
$169.95 KIT

FOR 4MHZ
ADD $10



KIT FEATURES:
1. Addressable as four separate 4K Blocks.
2. ON BOARD BANK SELECT circuitry. (Cromemco Standard!). Allows up to 512K on line!
3. Uses 2114 (450NS) 4K Static Rams.
4. ON BOARD SELECTABLE WAIT STATES.
5. Double sided PC Board, with solder mask and silk screened layout. Gold plated contact fingers.
6. All address and data lines fully buffered.
7. Kit includes ALL parts and sockets.
8. PHANTOM is jumpered to PIN 67.
9. LOW POWER: under 1.5 amps TYPICAL from the +8 Volt Buss.
10. Blank PC Board can be populated as any multiple of 4K.

BLANK PC BOARD W/DATA-$33
LOW PROFILE SOCKET SET-$12
SUPPORT IC'S & CAPS-$19.95
ASSEMBLED & TESTED-ADD $35

OUR #1 SELLING
RAM BOARD!

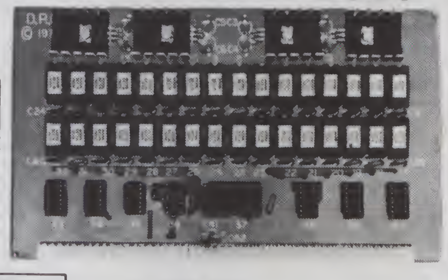
16K STATIC RAM SS-50 BUSS

PRICE CUT!

$159 KIT

FULLY STATIC!

FOR 2MHZ
ADD $10



FOR SWTPC
6800 BUSS!

ASSEMBLED AND
TESTED - $35

KIT FEATURES
1. Addressable on 16K Boundaries
2. Uses 2114 Static Ram
3. Fully Bypassed
4. Double sided PC Board. Solder mask and silk screened layout
5. All Parts and Sockets included
6. Low Power: Under 1.5 Amps Typical

BLANK PC BOARD—$35 COMPLETE SOCKET SET—$12
SUPPORT IC'S AND CAPS—$19.95

STEREO!
NEW! **S-100 SOUND COMPUTER BOARD** NEW!

At last, an S-100 Board that unleashes the full power of two unbelievable General Instruments AY3-8910 NMOS computer sound IC's. Allows you under total computer control to generate an infinite number of special sound effects for games or any other program. Sounds can be called in BASIC, ASSEMBLY LANGUAGE, etc.

KIT FEATURES:
* TWO GI SOUND COMPUTER IC'S.
* FOUR PARALLEL I/O PORTS ON BOARD.
* USES ON BOARD AUDIO AMPS OR YOUR STEREO.
* ON BOARD PROTO TYPING AREA.
* ALL SOCKETS, PARTS AND HARDWARE ARE INCLUDED.
* PC BOARD IS SOLDERMASKED, SILK SCREENED, WITH GOLD CONTACTS.
* EASY, QUICK, AND FUN TO BUILD, WITH FULL INSTRUCTIONS.
* USES PROGRAMMED I/O FOR MAXIMUM SYSTEM FLEXIBILITY.

Both Basic and Assembly Language Programming examples are included.

COMPLETE KIT!
$84.95
(WITH DATA MANUAL)

BLANK PC
BOARD W/DATA
$31

SOFTWARE:

SCL™ is now available! Our Sound Command Language makes writing Sound Effects programs a SNAP! SCL™ also includes routines for Register-Examine-Modify, Memory-Examine-Modify, and Play-Memory. SCL™ is available on CP/M* compatible diskette or 2708 or 2716. Diskette - **$24.95** 2708 - **$19.95** 2716 - ~~**$29.95**~~. Diskette includes the source. EPROM'S are ORG at E000H. **(Diskette is 8 Inch Soft Sectored)**

SPECIAL PURCHASE!

UART SALE!

TR1602B — SAME AS TMS6011, AY5-1013, ETC. 40 PIN DIP

TR1602B

$2.95 EACH **4 For $10.00**

CRT CONTROLLER CHIP

SMC #CRT 5037. PROGRAMMABLE FOR 80 x 24, ETC. VERY RARE SURPLUS FIND. WITH PIN OUT. $12.95 EACH.

4K STATIC RAM

National Semi. MM5257. Arranged 4K x 1. +5V, 18 PIN DIP. A Lower Power, Plug in Replacement for TMS 4044. 450 NS. Several Boards on the Market Will Accept These Rams. SUPER SURPLUS PURCHASE! **PRIME NEW UNITS!**

8 FOR $16 32 FOR $59.95

NEW! G.I. COMPUTER SOUND CHIP

AY3-8910. As featured in July, 1979 BYTE! A fantastically powerful Sound & Music Generator. Perfect for use with any 8 Bit Microprocessor. Contains 3 Tone Channels, Noise Generator, 3 Channels of Amplitude Control, 16 bit Envelope Period Control, 2-8 Bit Parallel I/O, 3 D to A Converters, plus much more! All in one 40 Pin DIP. Super easy interface to the S-100 or other busses. **$11.95** **PRICE CUT!**

SPECIAL OFFER: ~~$14.95~~ **each** Add $3 for 60 page Data Manual.

Digital Research Computers
(OF TEXAS)
P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 271-3538

TERMS: Add $2.00 postage. We pay balance. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCharge. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H. Orders over $50, add 85¢ for insurance.

ALL SALES ARE MADE SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. A COPY OF THIS WARRANTY IS AVAILABLE FREE, ON REQUEST.

*TRADEMARK OF DIGITAL RESEARCH.

WE ARE NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE SUPPLIERS OF CPM SOFTWARE.

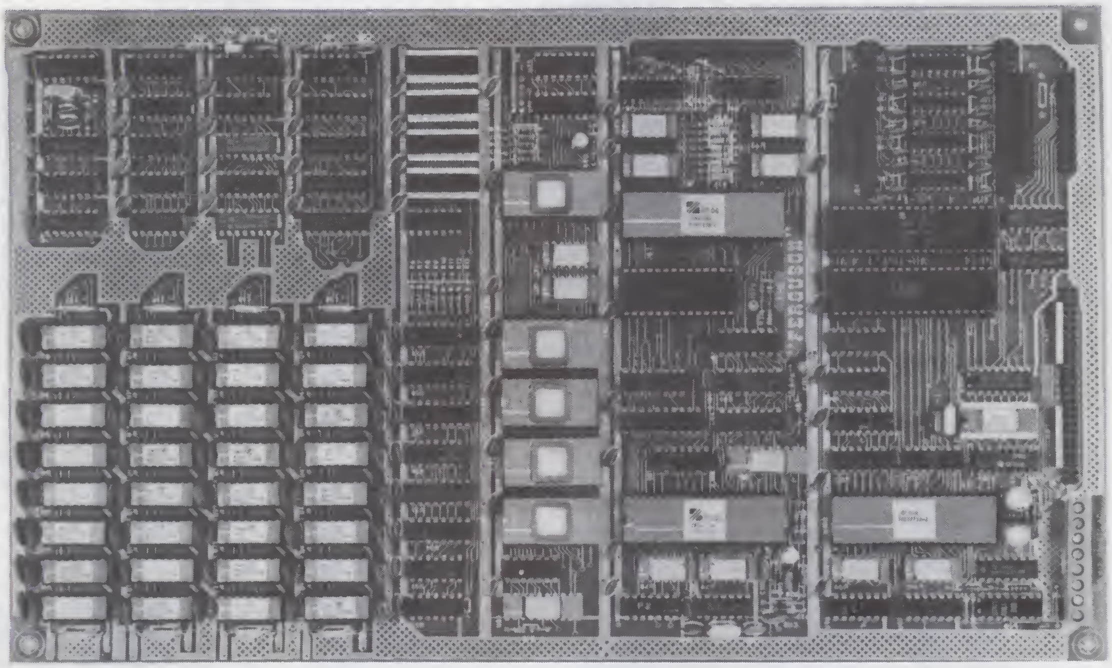
NEW! "THE BIG BOARD" NEW!

OEM - INDUSTRIAL - BUSINESS - SCIENTIFIC

SINGLE BOARD COMPUTER KIT!

Z-80 CPU! 64K RAM!

PARTIALLY ASSEMBLED KITS
For All Sockets Installed
And Soldered Add $50.



U.K. AND EUROPE: CONTACT VINCELORD LTD. 47 GOLDHURST TERRACE LONDON NW6

THE FERGUSON PROJECT: Three years in the works, and maybe too good to be true. A tribute to hard headed, no compromise, high performance, American engineering! The Big Board gives you all the most needed computing features on one board at a very reasonable cost. The Big Board was designed from scratch to run the latest version of CP/M*. Just imagine all the off-the-shelf software that can be run on the Big Board without any modifications needed! Take a Big Board, add a couple of 8 inch disc drives, power supply, an enclosure, C.R.T., and you have a total Business System for about 1/3 the cost you might expect to pay.

$649.00 ** (64K KIT BASIC I/O)

SIZE: 8½ x 13¾ IN.
SAME AS AN 8 IN. DRIVE.
REQUIRES: +5V @ 3 AMPS
+ - 12V @ .5 AMPS.

FULLY SOCKETED!

FEATURES: (Remember, all this on one board!)

64K RAM

Uses industry standard 4116 RAM'S. All 64K is available to the user, our VIDEO and EPROM sections do not make holes in system RAM. Also, very special care was taken in the RAM array PC layout to eliminate potential noise and glitches.

Z-80 CPU

Running at 2.5 MHZ. Handles all 4116 RAM refresh and supports Mode 2 INTERUPTS. Fully buffered and runs 8080 software.

SERIAL I/O (OPTIONAL)

Full 2 channels using the Z80 SIO and the SMC 8116 Baud Rate Generator. FULL RS232! For synchronous or asynchronous communication. In synchronous mode, the clocks can be transmitted or received by a modem. Both channels can be set up for either data-communication or data-terminals. Supports mode 2 Int. Price for all parts and connectors: $85.

BASIC I/O

Consists of a separate parallel port (Z80 PIO) for use with an ASCII encoded keyboard for input. Output would be on the 80 x 24 Video Display.

24 x 80 CHARACTER VIDEO

With a crisp, flicker-free display that looks extremely sharp even on small monitors. Hardware scroll and full cursor control. Composite video or split video and sync. Character set is supplied on a 2716 style ROM, making customized fonts easy. Sync pulses can be any desired length or polarity. Video may be inverted or true. 5 x 7 Matrix - Upper & Lower Case

FLOPPY DISC CONTROLLER

Uses WD1771 controller chip with a TTL Data Separator for enhanced reliability. IBM 3740 compatible. Supports up to four 8 inch disc drives. Directly compatible with standard Shugart drives such as the SA800 or SA801. Drives can be configured for remote AC off-on. Runs CP/M* 2.2.

TWO PORT PARALLEL I/O (OPTIONAL)

Uses Z-80 PIO. Full 16 bits, fully buffered, bi-directional. User selectable hand shake polarity. Set of all parts and connectors for parallel I/O: $29.95

REAL TIME CLOCK (OPTIONAL)

Uses Z-80 CTC. Can be configured as a Counter on Real Time Clock. Set of all parts: $14.95

SYSTEM COMPARISON

64K RAM KIT	$370.00
80 x 24 Video Kit	365.00
Floppy Disk Controller Kit	235.00
Z-80 CPU Kit	185.95
SER & PAR. I/O	129.95
S-100 Mother Board	45.00
SUB TOTAL	$1330.90

Talk about bangs per buck! The prices shown for S100 kits were taken from the July 1980 BYTE. This will give some basis for comparison between the Big Board and a similar system implementation on the S100 Buss.

CP/M* 2.2 FOR *BIG BOARD*

The popular CP/M* D.O.S. modified by MICRONIX SYSTEMS to run on Big Board is available for $150.00.

PC BOARD

Blank PC Board with Rom Set and Full Documentation. $199.00

PFM 3.0 2K SYSTEM MONITOR

The real power of the Big Board lies in its PFM 3.0 on board monitor. PFM commands include: Dump Memory, Boot CP/M*, Copy, Examine, Fill Memory, Test Memory, Go To, Read and Write I/O Ports, Disc Read (Drive, Track, Sector), and Search. PFM occupies one of the four 2716 EPROM locations provided.

Z-80 is a Trademark of Zilog.

Digital Research Computers
(OF TEXAS)

P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 271-3538

TERMS: Shipments will be made approximately 3 to 6 weeks after we receive your order. VISA, MC, cash accepted. We will accept COD's (for the Big Board only) with a $75 deposit. Balance UPS COD. Add $3.00 shipping.

USA AND CANADA ONLY

*TRADEMARK OF DIGITAL RESEARCH. NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE ORIGINATORS OF CPM SOFTWARE
**1 TO 4 PIECE DOMESTIC USA PRICE.

COMPONENTS

SN7400N	18	SN7462N	30
SN7402N	22	SN7492N	49
SN7404N	21	SN7493N	45
SN7408N	22	SN7495N	50
SN7410N	18	SN7496N	89
SN7412N	20	SN74122N	39
SN7413N	22	SN74136N	95
SN7414N	29	SN74141N	69
SN7416N	27	SN74151N	65
SN7417N	29	SN74153N	55
SN7420N	17	SN74154N	1 25
SN7425N	20	SN74155N	75
SN7430N	17	SN74157N	58
SN7437N	26	SN74160N	89
SN7438N	24	SN74161N	65
SN7440N	18	SN74163N	85
SN7442N	45	SN74164N	87
SN7443N	42	SN74165N	55
SN7445N	64	SN74174N	[illegible]
SN7451N	19	SN74175N	79
SN7454N	19	SN74180N	75
SN7474N	27	SN74181N	1 15
SN7475N	35	SN74393N	1 65

74LS00

74LS00	28	74LS158	89
74LS02	28	74LS161	83
74LS03	28	74LS162	89
74LS04	28	74LS163	98
74LS05	22	74LS164	85
74LS08	29	74LS165	85
74LS09	28	74LS169	1.55
74LS10	26	74LS170	1 75
74LS14	89	74LS174	85
74LS20	22	74LS175	85
74LS21	26	74LS190	85
74LS26	40	74LS191	1 25
74LS27	27	74LS195	95
74LS28	37	74LS197	78
74LS30	29	74LS221	1 25
74LS32	.31	74LS240	1 65
74LS38	31	74LS241	1 65
74LS42	63	74LS243	1 55
74LS48	77	74LS244	1 55
74LS74	38	74LS245	2 45
74LS75	55	74LS251	1 25
74LS86	45	74LS253	85
74LS90	59	74LS257	85
74LS93	65	74LS259	1 95
74LS96	80	74LS260	55
74LS107	43	74LS273	1 55
74LS113	45	74LS279	45
74LS122	45	74LS290	1 25
74LS123	89	74LS293	1 65
74LS125	89	74LS365	85
74LS126	79	74LS367	.75
74LS138	64	74LS373	1 45
74LS139	59	74LS374	1 45
74LS151	49	74LS377	1 25
74LS153	49	74LS669	1 55
74LS157	69	74LS670	1 85

74S00

74S00	39	74S138	75
74S02	45	74S140	1 00
74S03	38	74S158	75
74S04	39	74S174	1 35
74S05	39	74S175	1 35
74S10	39	74S182	75
74S15	45	74S189	4 25
74S20	55	74S201	6 75
74S22	55	74S240	2 75
74S30	75	74S244	2 95
74S37	55	74S251	2 75
74S50	65	74S287	2 95
74S51	49	74S288	2 95
74S64	55	74S299	5 75
74S74	65	74S470	9 25
74S86	95	74S471	9 50
74S112	1 95	74S473	9 50
74S132	1.45	74S474	9 50

EPROMS

2716 6 95 ea 6 for 6 25 ea
5V 450NS
TMS2716
2708 8 25 ea 8 for 7 50 ea
450NS 4 10 ea 8 for 3 95 ea

2732 19 95 ea 4 for 16 95 ea

4116 300NS 2 00 ea 8 for 14 00
200NS 2.35 ea 8 for 16 00

2114L 300NS 2.95 ea 4 for 2 75 ea
200NS 3 50 ea 4 for 3 00 ea

2111 450NS 2 50 ea 10 for 2 00 ea

CPUs

Z 80	7 95
Z 80A CTC	10 50
Z 80A CPU	10 50
Z 80 002 16 64K	129 00
8085A	13 50
2901A	7 50
MC6800	9 50

MISC

2102 450NS	95		
8038	2 95	1103A	75
NE555	.27	UPD765 floppy disk controller	17 85 w spe
AY5 1013A	4 25		
1488	1 10		
1489	1 10	ULN2001	1 95
8T26	1.30	TMS4400	1 40
8T28	1 30	MC4008P	1 75
8212	2.75	MH0026	1 55
8216	2.75	D3624	1 95
IS410SCR	95	D3001	1 95
IT410TRIAC	95	D3002	1 95
7905	85		
7908	85		
7915	85		
7918	85		
7805	85		
7806	85		
7808	85		
7812	85		
MC1330A1P	1 60		
MC1350P	1 15		
MC1358P	1 50		
LM380	1 20		
LM565N	95		
LM741	25		
MC1458P	55		
LM720	30		
LM386	1 75		

IC SOCKETS

6 PIN	10/1 20
14	10/1 40
16	10/1 60
18	10/1 90
20	10/2 80
22	10/2 80
24	10/2 80
28	10/3 80
40	10/4 75

"1981"
I.C. Master
$ 59.95

"GLOBAL" Logic Probes

LP–1: $40.00
Pulse width to 50nsec Rep. rates to 10mhz.

LP–3: $60.00
Pulse width to 6nsec. Rep. to over 60MHz.

LPK–1:
Logic Probe Kit - complete nothing extra to buy. Min. pulse width 300nsec.
$18.95

CARDS

MICROSOFT:
Z80 $295.00
16K RAM $160.00

VIDEX:
VIDEOTERM 80 column $295.00
KEYBOARD ENHANCER $120.00

CALIF. COMP. SYS:
APPLE CLOCK $124.00
PROTO BOARD $25.00

PRINTERS

EPSON:
MX–80 $560.00

CENTRONICS:
737–1 $950.00
753–1 $2,995.00

BASE II, inc:
800B $649.00
850 $749.00

Apple Expansion Kit
16K Memory Add-On
Includes Instructions
$25 95

TRS80
16K Add-On
Instructions &
Dip Switches
$25 95

SPECIALS

3inch MUFFIN FAN w/line cord!
$9.95

UPD 765C: Floppy Disk Controller
$17.95 w/spec's

2732A: 250NSEC. EPROM
$24.50

AY5–1013A: 30K Band UART
$3.95

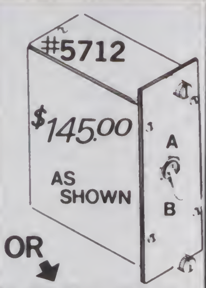
REAL–TIME CLOCK CALENDAR (MSM 5832)

Description: Mono. Metal Gate CMOS I.C.

Features
- Time, Month, Date, Year, & Day of Week
- Bus Oriented
- 4 Bit Data Bus
- 4 Bit Address
- R/W Hold Selec.+
- Inter. Signal
- 32.768Khz. xtal Control.
- 5v Pow. Sup.
- Low Power Dissipation

$7.45 W/SPEC's

XTAL $3.85



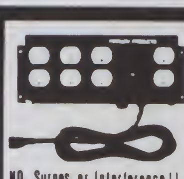
$145.00
AS SHOWN

OR

SHELF / DESK MOUNTED #5722
$155.00

DESC.
RS-232 COMPATABLE DIGITAL TRANSFER SWITCH WILL SWITCH MODEMS BETWEEN PROCESSORS / PRINTERS. 24 PINS ARE SWITCHED, PIN 1 WIRED TO GROUND.

T-bar INCORPORATED



NO Surges or Interference!!
THE MPD 117
turns an ordinary outlet into a controlled power source
$79.50

POWER SUPPLY

MODEL #CP198
input→110/125v
output→5vdc
At 6amps
$29.95
Qty. price avail.

CONCORD COMPUTER PRODUCTS

1971 SO. STATE COLLEGE
ANAHEIM, CALIF. 92806
(714) 937-0637

VISA · master charge · CHECK — M/O
NO COD

☆ send $1.00 for "81" catalog ☆

$10. MIN ORDER / CA. RES. ADD 6%
FRT.
[illegible] 8.00
[illegible] 10.00
[illegible] Inquire

COMPUTERS

ATARI® 800™ COMPUTER SYSTEM

400 Computer 8K → $350.00
800 Computer 16K → $759.00

*800 COMPUTER w/48K → $898.00

*Best Buy

ATARI PERIPHERALS:

Printer ("825")	$775.00	Printer ("822")	$379 00
Recorder	$65.00	Disk Drive	$565 00
Interface (850)	$175 00	Modem	$169 00
Paddles	$17 00	Joysticks	$17 00
Star Raiders	$49 00	Assembler Editor	$49 00
Space Invaders	$17 00	Music Composer	$49 00
Chess	$32 00	Mailing List	$17 00
Kingdom	$12 00	TV Switch Box	$8 95
Hangman	$12 00	16K RAM	$155 00
Blackjack	$12 00	8K RAM	$119 00

"APPLE II Plus"

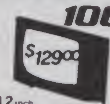
48k–$1199.00
64k–$1399.00

W/PURCHASE OF APPLE RECEIVE SPECIAL PRICING:

BASE II PRINTER → $619.00
DISK II w/cont. → $595.00
DISK II w/o cont. → $495.00
SUP'R'MOD II → $23.00
MONITOR: B&W
12 – 12MHz. → $125.00
INTEGER BASIC CARD → → $169.00

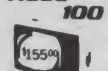
MONITORS

video 100



12 inch
12 MHz
BLACK & WHITE VIDEO MONITOR
AMDEK (LEEDEX) Corp.

video 100



12 inch 12MHz "AMDEK" GREEN Phos VIDEO MONITOR

$365.00

TERMINALS

SOROC TECHNOLOGY INC

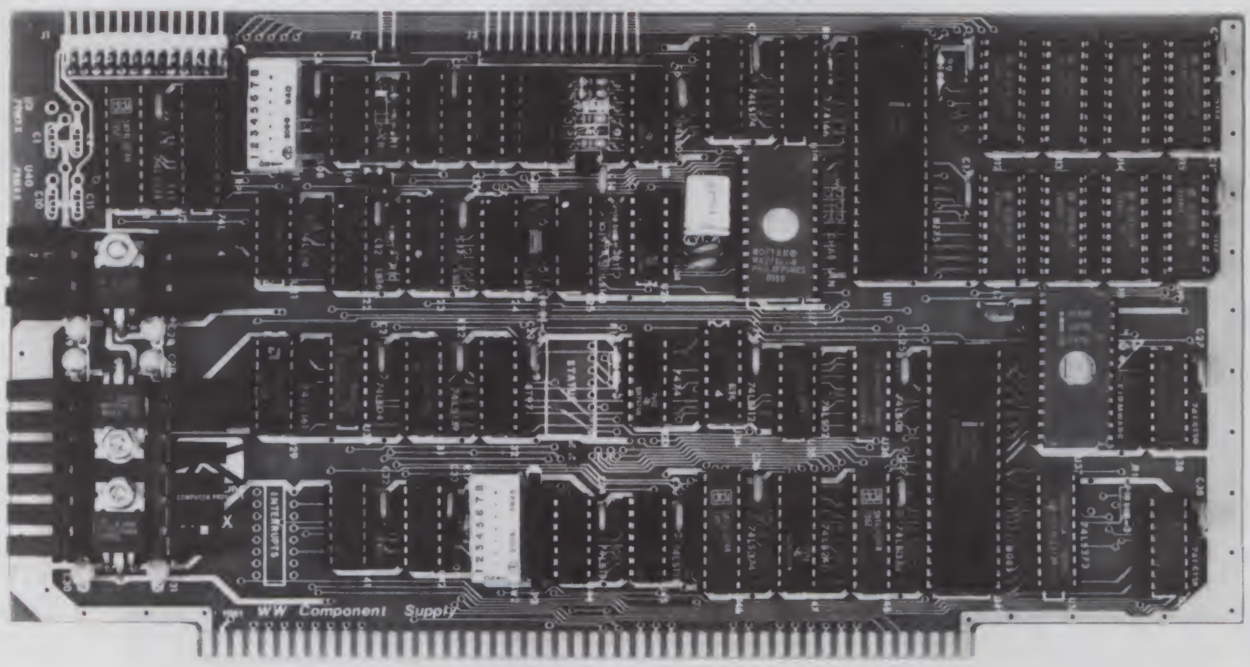
IQ120
$689.00

Synertek "KTM-3"
LOW COST TERMINAL WITH KEYBOARD AND VIDEO 100
→ $499.00 ←

See List of Advertisers on page 210

✓See List of Advertisers on page 210

INTELLIGENT VIDEO I/O FOR S-100 BUS



VIO-X

The VIO-X Video I/O Interface for the S-100 bus provides features equal to most intelligent terminals both efficiently and economically. It allows the use of standard keyboards and CRT monitors in conjunction with existing hardware and software. It will operate with no additional overhead in S-100 systems regardless of processor or system speed.

Through the use of the Intel 8275 CRT controller with an onboard 8085 processor and 4k memory, the VIO-X interface operates independently of the host system and communicates via two ports, thus eliminating the need for host memory space. The screen display rate is effectively 80,000 baud.

The VIO-X1 provides an 80 character by 25 line format (24 lines plus status line) using a 5 × 7 character set in a 7 × 10 dot matrix to display the full upper and lower case ASCII alphanumeric 96 printable character set (including true descenders) with 32 special characters for escape and control characters. An optional 2732 character generator is available which allows an alternate 7 × 10 contiguous graphics character set.

The VIO-X2 also offers an 80 character by 25 line format but uses a 7 × 7 character set in a 9 × 10 dot matrix allowing high-resolution characters to be used. This model also includes expanded firmware for block mode editing and light pen location. Contiguous graphics characters are not supported.

Both models support a full set of control characters and escape sequences, including controls for video attributes, cursor location and positioning, cursor toggle, and scroll speed. An onboard Real Time Clock (RTC) is displayed in the status line and may be read or set from the host system. A checksum test is performed on power-up on the firmware EPROM.

Video attributes provided by the 8275 in the VIO-X include:

- FLASH CHARACTER
- INVERSE CHARACTER
- UNDERLINE CHARACTER or
- ALT. CHARACTER SET
- DIM CHARACTER

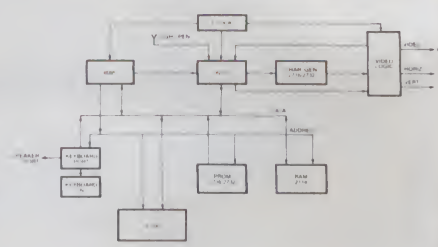
The above functions may be toggled together or separately.

The board may be addressed at any port pair in the IEEE 696 (S-100) host system. Status and data ports may be swapped if necessary. Inputs are provided for parallel keyboard and for light pen as well as an output for audio signalling. The interrupt structure is completely compatible with Digital Research's MP/M ®.

Additional features include:

- HIGH SPEED OPERATION
- PORT MAPPED IEEE S-100 INTERFACE
- FORWARD/REVERSE SCROLL or
- PROTECTED SCREEN FIELDS
- CONVERSATIONAL or BLOCK MODE (opt)
- INTERRUPT OPERATION
- CUSTOM CHARACTER SET
- CONTROL CHARACTERS
- ESCAPE CHARACTER COMMANDS
- INTELLIGENT TERMINAL EMULATION
- TWO PAGE SCREEN MEMORY

VIO-X1 – 80 × 25 5 ×7 A & T **$295.00**
Conversational Mode
VIO-X2 – 80 × 25 7 × 7 A & T **$345.00**
Conversational & Block Modes



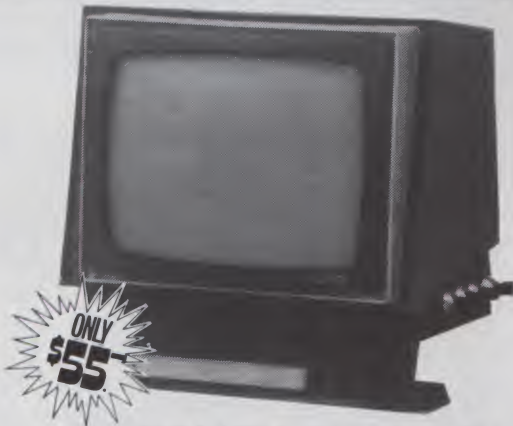
VIO-X S-100 I/O INTERFACE

FULCRUM™
COMPUTER PRODUCTS

Distributed by ✓285
WW COMPONENT SUPPLY INC. 1771 JUNCTION AVENUE • SAN JOSE, CA 95112 • (408) 295-7171

SELECTRONICS

Does it again. . . .high quality Sylvania monitors at the lowest prices ever. These monitors have been thoroughly checked and guaranteed.



MONITORS

12" black & white monitor. Wide band, will display 80 x 24 char. 10K or 75Ω input impedance, composite video input. Transformer power supply.
Shpt. Wt. 30#
Price: $55 ea.
Used with all computers such as: TRS-80, Apple, and many others.
Complete manual $5.00.

PERKIN-ELMER CAROUSEL 350 TERMINAL

Only Selectronics could bring you this unbelievable opportunity to own the finest high-quality KSR terminal ever made. These machines originally listed for more than $4200. Check these outstanding features:

* Letter quality printing with upper and lowercase
* Microprocessor controlled
* 76-key alphanumeric keyboard with 10-key numeric pad
* EIA RS232C asynchronous serial interface
* 40 cps print rate
* 132 character buffer
* Full or half duplex
* Parity—odd, even, none
* Replacable print cups and elements

Excellent Condition—$750.00 F.O.B. our warehouse.
Shpt. Wt. 90#

HAZELTINE 1200 TERMINALS

24 x 80 character display; 5x7 dot matrix, Raster Scan, full ASCII keyboard; dc to 9600 baud; RS-232 interface; odd, even or no parity enable; full/half duplex, MOS-shift register memory with constant refresh.

ONLY $200.00

Also HAZELTINE 1000's

12 x 80 character baud rate 300
Shpt. Wt. 45#
FOB our warehouse
ONLY $150.00

ACOUSTIC COUPLE MODEM

Originate only, 300 baud, full or half duplex.
SHPT. WT. 10#
ONLY $49.00

COMPUTER CARBONLESS PAPER

14-7/8 x 11, one-part. 3000 sheets per carton.
$24.00 per carton, 5 for $100.00
FOB Phila., Pa.
Shipping Weight 35 lbs.

DIGITAL DISPLAY BOARDS

6 digit numeric display boards with 6 FND 507. Common anode displays and 10 red LED's. With drivers & logic for multiplexed operation. Price: $5.00 ea. or 6/$25.00.

All items have been thoroughly checked and are in excellent working condition.

* Test Equipment
* Power Supply Components
* Power Supplies
* Communication Equipment
* Pulse Equipment

✓205

SELECTRONICS

Phone:
215-468-4645
215-468-7891

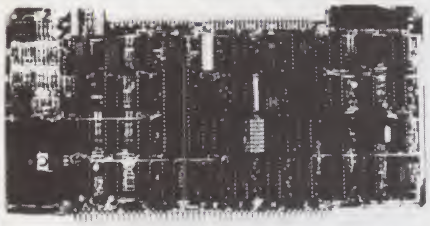
1206 S. Napa Street • Philadelphia PA 19146

Penna. resident please add 6% sales tax. **ALL PRICES ARE F.O.B.** our warehouse. Phila. PA. All merchandise accurate as to description to the best of our knowledge. Your purchase money refunded if not satisfied. Min. order $10.00.

Better Products,

JADE Double-D ™

Get the Inside Track



S-100 bus compatible • Reads and writes single or double density • Density is software selectable • Controls up to four 5-1/4" or 8", single or double-sided drives • Single or double-sided drives may be mixed in the same system • On-board Z-80A to assure reliable operation • EIA level serial printer interface on board, baud rates to 9600 (perfect for despooling operations) • Uses IBM standard formats • Designed to meet IEEE signal disciplines • Works with 8080, 8085, and Z-80 CPU's.

IOD-1200B	*Bare Board*	**$59.95**
IOD-1200K	*Kit*	**$299.95**
IOD-1200A	*A & T*	**$375.00**
SFC-58001200E	*DD boot PROM*	**$20.00**
SFC-59002001F	*CP/M 2.2 for DD*	**$149.95**

JADE Memory Bank ™

8 or 16 Bit Dynamic Memory

A 64K S-100 dynamic memory that looks toward the future. • S-100 standard pinout and signal discipline • 8 or 16 bit words, automatically, depending on the type of CPU on the bus • 4-layer PC board designed for extremely low-noise operation.

With its on-board refresh controller, this board allows DMA operations without regard to time factors. It will run reliably at any system clock rate up to 4 MHz--because it is clocked with the system clock itself (no one-shots are used for timing operations). Its unique design enables it to switch from a 64K by 8-bit board to a 32K by 16-bit board automatically (responds to IEEE's 16 Rqst line and, if enabled, replies with 16 Grant).

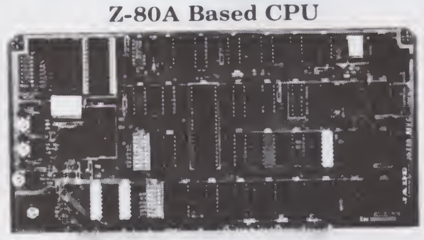
MEM-99730B	*Bare Board*	**$49.95**
MEM-99730K	*Kit, no RAM*	**$199.95**
MEM-16730K	*16K kit*	**$219.95**
MEM-32731K	*32K kit*	**$239.95**
MEM-48732K	*48K kit*	**$259.95**
MEM-64733K	*64K kit*	**$279.95**
Assembled & Tested		*add* **$50.00**

4 MHz Z-80 CPU, 80 x 25 display with graphics and 8 colors, 32K RAM, 24K ROM, parallel/serial/cassette interfaces, upper/lower case, numeric keypad, 10 special function keys, uses CP/M 2.2.

NEC-8001A	*32K CPU/keyboard*	**$1095.00**
NEC-8012A	*I/O with 32K RAM*	**$695.00**
NEC-8031A	*Dual disk unit*	**$1095.00**
VDM-651200	*12" green CRT*	**$269.95**
NEC-1202D	*Hi-res RGB color CRT*	**$1045.00**
NEC-90005	*Gen. accting softwr*	**$375.00**

JADE The Big Z ™

Z-80A Based CPU



S-100 bus compatible • Switch selectable 2 or 4 MHz operation • Serial I/O port • Accomodates a 2708, 2716, or 2732 EPROM in shadow mode allowing full use of 64K RAM • MWRITE signal is generated automatically if used without front panel • On-board 8251 USART controls serial port at baud rates from 75 to 9600 baud • Switch selectable 1K, 2K, or 4K boundary fully buffered • Power-on jump to EPROM

CPU-30201K	*Kit*	**$139.95**
CPU-30201A	*A & T*	**$189.95**
CPU-30200B	*Bare board*	**$35.00**

Memory Chips on Sale

	1-9	10-24	25-49	50+
2114L *4 MHz*	3.35	2.99	2.75	2.49
2708 *2 MHz*	4.90	3.90	3.45	2.90
2532 *2 MHz*	24.90	19.90	15.90	12.90
2716 *2 MHz*	8.90	7.45	6.45	5.75
2716 *4 MHz*	19.90	15.45	13.45	11.75
2732 *2 MHz*	24.90	19.90	15.90	12.90
2732 *4 MHz*	39.90	29.90	24.90	19.90
2758 *2 MHz*	6.90	6.25	5.50	4.50
4116 *200 ns*	3.25	2.99	2.49	1.99
4164 *200 ns*	28.90	24.90	22.90	19.90

JADE Disk Sub-Systems

Shugart, Siemens, Qume

Handsome metal cabinet with proportionally balanced air flow system • Rugged dual drive power supply • Power cable kit • Power switch, line cord, fuse holder, cooling fan • Never-Mar rubber feet • All necessary hardware to mount 2-8" disk drives, power supply, and fan • Does not include signal cable

Dual 8" Subassembly Cabinet

END-000420	*Bare cabinet*	**$59.95**
END-000421	*Cabinet kit*	**$225.00**
END-000431	*A & T*	**$359.95**

8" Disk Drive Subsystems
Single Sided, Double Density

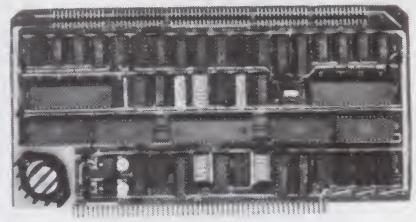
END-000423	*Kit w/ 2 FD100-8Ds*	**$975.00**
END-000424	*A & T w/ 2 FD100-8Ds*	**$1175.00**
END-000433	*Kit w/ 2 SA-801Rs*	**$999.95**
END-000434	*A & T w/ 2 SA-801Rs*	**$1195.00**

8" Disk Drive Subsystems
Double Sided, Double Density

END-000426	*Kit w/ 2 DT-8s*	**$1475.00**
END-000427	*A & T w/ 2 DT-8s*	**$1675.00**
END-000436	*Kit w/ 2 SA-851Rs*	**$1495.00**
END-000437	*A & T w/ 2 SA-851Rs*	**$1695.00**

JADE S.P.I.C.

Our "SPICy" New I/O Board



New, from JADE, one of the most advanced, technologically sophisticated Serial/Parallel Interrupt Controller systems in the world. On a single IEEE S-100 standard board, JADE has packed two bi-directional parallel ports with full handshaking, four serial channels (asynchronous, IBM-compatible bi-synch, synchronous, HDLC/SDLC) with full and complete modem control lines, and 16 counter-timer channels.

Utilizing the highly advanced Zilog peripheral chips, (Z-80 SIO, PIO & CTCs), the SPIC board is fully programmable to serve as the foundation for a multi-user multi-tasking system. Each of the seven Z-80 peripheral chips can generate its own interrupt vector, with daisy-chain priority levels. Each counter-timer channel can be programmed to monitor an interrupt vector line on the S-100 bus. Each SIO channel can be driven independently with separate Tx/Rx clocks so your peripherals can have varied baud rates from 110 to 76,800 baud.

IOI-1045B	*Bare Board*	**$49.95**
IOI-1045K	*2 CTC's, 1 SIO, 1 PIO*	**$179.95**
IOI-1045A	*A & T*	**$239.95**
IOI-1046K	*4 CTC's, 2 SIO's, 1 PIO*	**$219.95**
IOI-1046A	*A & T*	**$299.95**

Build It Yourself and Save $1850

Complete S-100 Micro-Computer Kit

4 Mhz Big Z CPU, 64K Memory Bank, Double-D disk controller, 12 slot mainframe, dual disk sub-assembly with 2 double density 8" disk drives & power supply, ADDS Viewpoint terminal, CP/M 2.2, boot PROM, system monitor, & all necessary cables and manuals - ***PLUS FREE Business Software Package.*** *Save over $1850.00* **$2795.00**

Jade Business Software

Now, for the first time, at an amazingly low price, CP/M and SDOS users can get an affordable business software package. Just look at these programs! All come complete with basic code (written to run under CBASIC Ŧ), for easy customizing and modification. All software is self-documenting. The package price includes a word-processing system. When printed on the system's printer, all files with the extension xxxTOT. constitute a complete manual. No manuals are supplied with the package other than as they appear on the disks. All software is supplied on 8" premium quality JADE Diskettes. Software sold as is. (Sorry, but at these low prices we can not offer our usual friendly support and handholding.)

Includes; General Ledger, Payroll, Invoicing, Inventory Control, Accounts Recievable, Cash Disbursements, Fixed Assets, Cash Receipts, & Mailing List Management.

Special Package Price ***$249.95***

Lower Prices,

Printers

BEST BUY in PRINTERS - Epson

MX-70 *132 column, 80 CPS, 5 x 7 dot matrix, adjustable tractor feed, & graphics*

PRM-27070 *List $459* **$399.95**

MX-80 *132 column, 80 CPS, bi-directional/logic seeking printing, 9 x 9 dot matrix, adjustable tractor feed, & 64 graphics characters*

PRM-27080 *List $645* **$474.95**

MX-80FT *same as MX-80 with friction feed and full graphics added*

PRM-27082 *List $745* **$574.95**

MX-100 *233 column, correspondence quality, ultra-high resolution graphics, up to 15" paper, friction feed & removable adjustable tractor feed, 18 x18 dot matrix, 80 CPS, programmable forms handling*

PRM-27100 *List $945* **$795.00**

PRA-27084 *Serial interface* **$69.95**
PRA-27088 *Serial intf & 2K buffer* .. **$144.95**
PRA-27081 *Apple card* **$74.95**
PRA-27082 *Apple cable* **$22.95**
PRA-27086 *IEEE 488 card* **$52.95**
PRA-27087 *TRS-80 cable* **$32.95**
PRA-27085 *Graftrax II* **$95.00**
PRA-27083 *Extra ribbon* **$14.95**

SPINWRITER - NEC

65 cps, bi-directional, letter quality printer with deluxe tractor mechanism, both parallel and serial interfaces on-board, 16K buffer, ribbon, print thimble, graphics, micro-space justification, data cable, and self test diagnostic ROM.

PRD-55511 *without 16K buffer* ... **$2795.00**
PRD-55512 *with 16K buffer* **$2895.00**

Accessories for TRS-80

DISK DRIVES for TRS-80

23% more storage, 8 times faster, 40 track with free patch, 120 day warranty.

MSM-12410C *Save $125.00 !!!* **$325.00**

8" DISK DRIVES for MODEL II

2 double density drives with cabinet, power supply, & cables

END-000433 *Kit* **$1050.00**
END-000434 *Assembled* **$1250.00**
WCA-5036A *Cable (required)* **$29.95**

16K Atari ... $359.95

ATARI 800 - Atari

Complete personal computer with high resolution color graphics, built-in RF modulator, 4 controller ports, internal speaker, 16K RAM & 8K ROM

SYO-2080A *16K Atari 800* **$759.95**
Atari 800 with 32K of RAM **$799.95**
Atari 800 with 48K of RAM **$849.95**
SYO-2040A *16K Atari 400* **$359.95**
MSM-330810 *Disk drive* **$595.00**
MSM-330815 *Dual drive* **$1395.00**
IOX-5050A *850 interface* **$199.95**
MEX-16853K *16K RAM module* **$69.95**
SFI-241011005 *Visicalc* **$184.95**

Accessories for Apple

16K MEMORY UPGRADE

Add 16K of RAM to your TRS-80, Apple, or Exidy in just minutes. We've sold thousands of these 16K RAM upgrades which include the appropriate memory chips (as specified by the manufacturer), all necessary jumper blocks, fool-proof instructions, and our 1 year guarantee.

MEX-16100K *TRS-80 kit* **$25.00**
MEX-16101K *Apple kit* **$25.00**
MEX-16102K *Exidy kit* **$25.00**

16K RAM Card - Microsoft

(There is life after 48K)

MEX-16300A *A & T* **$174.95**

Z-80* CARD for APPLE

Two computers in one, Z-80 & 6502, more than doubles the power & potential of your Apple, includes Z-80 CPU card, CP/M 2.2, & BASIC-80*

CPX-30800A *A & T* **$299.95**

APPLE CLOCK - Cal Comp Sys

Real time clock w/ battery back-up

IOK-2030A *A & T* **$109.95**

DISK DRIVE for APPLE

5¼" disk drive with controller for your Apple

MSM-12310C *with controller* **$475.00**
MSM-123101 *w/out controller* **$375.00**

8" DRIVES for APPLE

Controller, DOS, two 8" double densisty drives, cabinet, power supply, & cables

Special Package Price *Kit* **$1399.95**

PRINTER INTERFACE - C.C.S.

Centronics type I/O card w/ firmware

IOI-2041A *A & T* **$99.95**

AIO, ASIO, APIO - S.S.M.

Parallel & serial interface for your Apple (see Byte pg 11)

IOI-2050K *Par & Ser kit* **$139.95**
IOI-2050A *Par & Ser A & T* **$169.95**
IOI-2052K *Serial kit* **$89.95**
IOI-2052A *Serial A & T* **$99.95**
IOI-2054K *Parallel kit* **$69.95**
IOI-2054A *Parallel A & T* **$89.95**

A488 - S.S.M.

IEEE 488 controller, uses simple basic commands, includes firmware and cable, 1 year guarantee, (see April Byte pg 11)

IOX-7488A *A & T* **$399.95**

CPS MULTICARD - Mtn. Computer

Three cards in one! Real time clock/calendar, serial interface, & parallel interface - all on one card.

IOX-2300A *A & T* **$199.95**

Apple-CAT - Novation

Software selectable 1200 or 300 baud, direct connect, auto-answer/auto-dial, auxiliary 3-wire RS232C serial port for printer.

IOM-5232A *Save $50.00!!!* **$339.95**

Single Board Computers

AIM-65 - Rockwell

6502 computer with alphanumeric display, printer, & keyboard, and complete instructional manuals

CPK-50165 *1K AIM* **$424.95**
CPK-50465 *4K AIM* **$499.95**
SFK-74600008E *8K BASIC ROM* ... **$99.95**
SFK-64600004E *4K assembler ROM* **$84.95**
PSX-030A *Power supply* **$64.95**
ENX-000002 *Enclosure* **$54.95**

4K AIM, 8K BASIC, power supply, & enclosure

Special package price **$675.00**

Z-80* STARTER KIT - SD Systems

Complete Z-80 computer with RAM, ROM, I/O, display, keyboard, manual, and kluge area.*

CPS-30010K *Kit* **$369.95**
CPS-30010A *A & T* **$459.95**

SYM-1 - Synertek Systems

Single board computer with 1K of RAM, 4K of ROM, key-pad, LED display, 20ma & cassette interface on board.

CPK-50020A *A & T* **$249.95**

Video Terminals

VIEWPIONT - ADDS

Detachable keyboard, serial RS232C interface, baud rates from 110 to 19,200, auxiliary serial output port, 24 x 80 display, tiltable screen, international character set

VDT-501210 *Sale Priced* **$639.95**

TELEVIDEO 950

Detachable keyboard, split screen with line lock, etched CRT, programmable function keys, on-screen status line, buffered auxiliary port, 14 x 10 dot matrix, self test, serviced nation-wide by General Electric

VDT-901250 *List $1195.00* **$995.00**

Video Monitors

Leedex / Amdek

Reasonably priced video monitors

VDM-801210 *Video 100 12" B&W* .. **$139.95**
VDM-801230 *Video 100-80 12" B& W* **$179.95**
VDM-801250 *12" Green Phospor* **$169.95**
VDC-801310 *13" Color I* **$379.95**

13" COLOR MONITOR - Zenith

The hi res color you've been promising yourself

VDC-201301 .... **$449.00**

12" GREEN SCREEN - NEC

20 MHz, P31 phosphor video monitor with audio, exceptionally high resolution - A fantastic monitor at a very reasonable price

VDM-651200 *12" monitor* **$269.95**

EPROM Erasers

EPROM ERASERS

L.S. Engineering UV eraser for up to 48 EPROMs

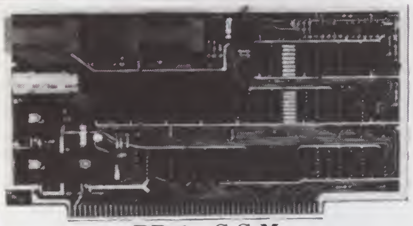
XME-3200 *A & T* **$39.99**

Spectronics hi intensity industrial eraser

XME-3100 *Without timer* **$69.50**
XME-3101 *With timer* **$94.50**

Faster Service.

S-100 PROM Boards



PB-1 - S.S.M.
2708, 2716 EPROM board with built-in programmer
MEM-99510K *Kit* $154.95
MEM-99510A *A & T* $219.95

PROM-100 - SD Systems
2708, 2716, 2732, 2758, & 2516 EPROM programmer
MEM-99520K *Kit* $219.95
MEM-99520A *Jade A & T* $269.95

EPROM BOARD - Jade
16K or 32K uses 2708's or 2716's, 1K boundary
MEM-16230K *Kit* $79.95
MEM-16230A *A & T* $119.95

S-100 Video

VB-3 - S.S.M.
80 characters x 24 lines expandable to 80 x 48 for a full page of text, upper & lower case, 256 user defined symbols, 160 x 192 graphics matrix, memory mapped, has key board input.
IOV-1095K *4 MHz kit* $349.95
IOV-1095A *4 MHz A & T* $439.95
IOV-1096K *80 x 48 upgrade* $39.95

VDB-8024 - SD Systems
80 x 24 I/O mapped video board with keyboard I/O, and on-board Z-80A.*
IOV-1020K *Kit* $399.95
IOV-1020A *Jade A & T* $459.95

VIDEO BOARD - S.S.M.
64 characters x 16 lines, 128 x 48 matrix for graphics, full upper/lower case ASCII character set, numbers, symbols, and greek letters, normal/reverse/blinking video, S-100.
IOV-1051K *Kit* $149.95
IOV-1051A *A & T* $219.95
IOV-1051B *Bare board* $34.95

Motherboards

ISO-BUS - Jade
Silent, simple, and on sale - a better motherboard

6 Slot (5¼" x 8⅝")
MBS-061B *Bare board* $19.95
MBS-061K *Kit* $39.95
MBS-061A *A & T* $49.95

12 Slot (9¾" x 8⅝")
MBS-121B *Bare board* $29.95
MBS-121K *Kit* $69.95
MBS-121A *A & T* $89.95

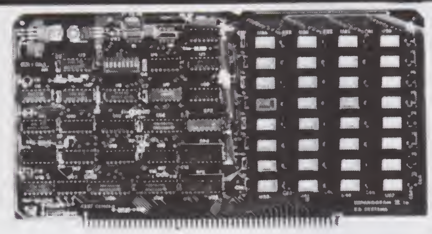
18 Slot (14½" x 8⅝")
MBS-181B *Bare board* $49.95
MBS-181K *Kit* $99.95
MBS-181A *A & T* $139.95

Mainframes

MAINFRAME - Cal Comp Sys
12 slot S-100 mainframe with 20 amp power supply
ENC-112105 *Kit* $379.95
ENC-112106 *A & T* $409.95

DISK MAINFRAME - N.P.C.
Holds 2 8" drives and a 12 slot S-100 system. Attractive metal cabinet with 12 slot motherboard & card cage, power supply, dual fans, lighted switch, and other professional features
ENS-112325 *with 25 amp p.s.* $699.95

S-100 Memory



EXPANDORAM II - S D Systems
4 MHz RAM board expandable from 16K to 64K
MEM-16630K *16K kit* $275.95
MEM-32631K *32K kit* $295.95
MEM-48632K *48K kit* $315.95
MEM-64633K *64K kit* $335.95
Assembled & tested .... add $50.00

64K RAM - Calif Computer Sys
4 MHz bank port / bank byte selectable, extended addressing, 16K bank selectable, PHANTOM line allows memory overlay, 8080 / Z-80 / front panel compatible.
MEM-64565A *A & T* $575.00

32K STATIC RAM - Jade
2 or 4 MHz expandable static RAM board uses 2114L's
MEM-16151K *16K 4 MHz kit* $169.95
MEM-32151K *32K 4 MHz kit* $299.95
Assembled & tested .... add $50.00

16K STATIC RAM - Mem Merchant
4 MHz 16K static RAM board, IEEE S-100, bank selectable, Phantom capability, addressable in 4K blocks, "disable-able" in 1K segments, extended addressing, low power
MEM-16171A *A & T* $174.95

S-100 Disk Controller

DOUBLE DENSITY - Cal Comp Sys
5¼" and 8" disk controller, single or double density, with on-board boot loader ROM, and free CP/M 2.2 and manual set.*
IOD-1300A *A & T* $369.95

VERSAFLOPPY II - SD Systems
New double density controller for both 8" & 5¼"
IOD-1160K *Kit* $339.95
IOD-1160A *A & T* $379.95

Disk Drives

8" DISK DRIVES
Highly reliable double density floppy disk drives

Shugart 801R single sided, double density
MSF-10801R *SA-801R* $425.00
***Special Sale Price** .... 2 for $790.00*

Shugart 851R double sided, double density
MSF-10851R *SA-851R* $595.00
***Special Sale Price** .... 2 for $1150.00*

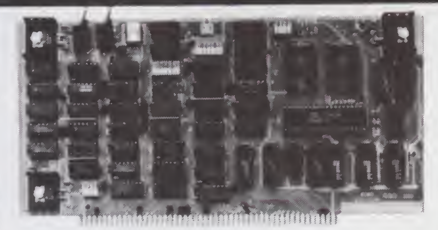
Siemens FDD100-8D2 single sided, double density
MSF-201120 ...... $385.00
***Special sale price** .... 2 for $750.00*

Qume Datatrak 8 double sided, double density
MSF-750080 *SA-851R compatible* .. $599.95
***Special sale price** .... 2 for $1160.00*

JADE DISK PACKAGE
Double density controller, two 8" double density floppy disk drives, CP/M 2.2 (configured for controller), hardware and software manuals, boot PROM, cabinet, power supply, fan, & cables
Special Package Price *Kit* $1395.00

5¼" Diskettes *Box of 10* .. $22.00
8" Diskettes *Box of 10* $24.00

S-100 CPU



CB-2 Z-80 CPU - S.S.M.
2 or 4 MHz Z-80 CPU board with provision for up to 8K of ROM or 4K of RAM on board, extended addressing, IEEE S-100, front panel compatible.
CPU-30300K *Kit* $239.95
CPU-30300A *A & T* $299.95

2810 Z-80* CPU - Cal Comp Sys
2/4 MHz Z-80A CPU with RS-232C serial I/O port and on-board MOSS 2.2 monitor PROM, front panel compatible.*
CPU-30400A *A & T* $269.95

SBC-200 - SD Systems
4 MHz Z-80 CPU with serial & parallel I/O ports, up to 8K of on-board PROM, software programmable baud rate generator, 1K of on-board RAM, Z-80 CTC.*
CPC-30200K *Kit* $339.95
CPC-30200A *Jade A & T* $399.95

S-100 I/O

I/O-4 - S.S.M.
2 serial I/O ports plus 2 parallel I/O ports
IOI-1010K *Kit* $179.95
IOI-1010A *A & T* $249.95
IOI-1010B *Bare board* $35.00

Novation Cat Modem

-Limited Quantity-

IOM-5200A *List price $189.95* $125.00

D-CAT 300 baud, direct connect modem
IOM-5201A *Special sale price* $169.95

AUTO-CAT Auto answer/origiate, direct connect
IOM-5230A *Special sale price* $239.95

Place Orders Toll Free

Continental U.S. 800-421-5500
Inside California 800-262-1710

For Technical Inquires or Customer Service call:
213-973-7707

JADE ✓48
Computer Products
4901 W. Rosecrans, Hawthorne, Ca 90250

TERMS of SALE: Cash, checks, credit cards, or Purchase Orders from qualified firms and institutions. **Minimum Order $15.00.** California residents add 6% tax. Minimum shipping & handling charge $3.00. **Pricing & availibility subject to change**

For further information from our advertisers, please use the Reader Service card.

*This advertiser prefers to be contacted directly.

READER SERVICE

Please help us to bring you a better magazine—by answering these questions:

I. What is your age?
- ☐ A 25 or under
- ☐ B 26-35
- ☐ C 36-45
- ☐ D 46-60
- ☐ E Over 60

II. What is your annual income?
- ☐ 1 Under $20,000
- ☐ 2 $20,000-$30,000
- ☐ 3 $31,000-$40,000
- ☐ 4 $41,000-$50,000
- ☐ 5 over $50,000

III. What is your education level?
- ☐ A High School
- ☐ B Trade or Technical School
- ☐ C College
- ☐ D Graduate School

IV. Do you own a microcomputer?
- ☐ 1 Yes
- ☐ 2 No

V. What kind of system?
- ☐ A Apple
- ☐ B Gimix
- ☐ C Heath
- ☐ D Hewlett-Packard
- ☐ E Other—Please specify____________
- ☐ F MSI
- ☐ G North Sta
- ☐ H OSI
- ☐ I PET
- ☐ J TRS-80

VI. What was the cost?
- ☐ 1 Under $1,000
- ☐ 2 $1,000-$2,000
- ☐ 3 $2,000-$4,000
- ☐ 4 More than $4,000

VII. What is the primary use of your computer?
- ☐ A Business (Income production)
- ☐ B Personal (Home Management and/or Entertainment
- ☐ C Educational
- ☐ D Scientific/Technical
- ☐ E Other____________

VIII. What is your occupational field?
- ☐ 1 Professional (Lawyer, Accountant, Physician, etc.)
- ☐ 2 Engineering
- ☐ 3 Business
- ☐ 4 Educational
- ☐ 5 Government
- ☐ 6 Consultant

IX. Does your employer own a computer?
- ☐ A Yes
- ☐ B No
- ☐ C Self-employed (See 7)

X. Do you work with the company computer?
- ☐ 1 Yes
- ☐ 2 No
- ☐ 3 N/A

XI. Do you have any influence on the use of the company computer?
- ☐ A Yes
- ☐ B No
- ☐ C N/A

XII. Do you have any influence over or make the decisions on the purchase of computer equipment for your employer?
- ☐ 1 Yes
- ☐ 2 No
- ☐ 3 N/A

XIII. If you are not a subscriber, please circle 500.

Reader Service: Return this card to receive full information on the products advertised in this issue. Refer to the ad. You will find numbers near the logo of each advertiser. Each represents the advertiser's individual Reader Service number. Circle the corresponding numbers on one of the cards on this page, include your name, address & zip, and drop in a mailbox. In 4-6 weeks you'll hear from the advertiser directly.

This card valid until October 31, 1981

1	6	11	16	21	126	131	136	141	146	251	256	261	266	271	376	381	386	391	396
2	7	12	17	22	127	132	137	142	147	252	257	262	267	272	377	382	387	392	397
3	8	13	18	23	128	133	138	143	148	253	258	263	268	273	378	383	388	393	398
4	9	14	19	24	129	134	139	144	149	254	259	264	269	274	379	384	389	394	399
5	10	15	20	25	130	135	140	145	150	255	260	265	270	275	380	385	390	395	400
26	31	36	41	46	151	156	161	166	171	276	281	286	291	296	401	406	411	416	421
27	32	37	42	47	152	157	162	167	172	277	282	287	292	297	402	407	412	417	422
28	33	38	43	48	153	158	163	168	173	278	283	288	293	298	403	408	413	418	423
29	34	39	44	49	154	159	164	169	174	279	284	289	294	299	404	409	414	419	424
30	35	40	45	50	155	160	165	170	175	280	285	290	295	300	405	410	415	420	425
51	56	61	66	71	176	181	186	191	196	301	306	311	316	321	426	431	436	441	446
52	57	62	67	72	177	182	187	192	197	302	307	312	317	322	427	432	437	442	447
53	58	63	68	73	178	183	188	193	198	303	308	313	318	323	428	433	438	443	448
54	59	64	69	74	179	184	189	194	199	304	309	314	319	324	429	434	439	444	449
55	60	65	70	75	180	185	190	195	200	305	310	315	320	325	430	435	440	445	450
76	81	86	91	96	201	206	211	216	221	326	331	336	341	346	451	456	461	466	471
77	82	87	92	97	202	207	212	217	222	327	332	337	342	347	452	457	462	467	472
78	83	88	93	98	203	208	213	218	223	328	333	338	343	348	453	458	463	468	473
79	84	89	94	99	204	209	214	219	224	329	334	339	344	349	454	459	464	469	474
80	85	90	95	100	205	210	215	220	225	330	335	340	345	350	455	460	465	470	475
101	106	111	116	121	226	231	236	241	246	351	356	361	366	371	476	481	486	491	496
102	107	112	117	122	227	232	237	242	247	352	357	362	367	372	477	482	487	492	497
103	108	113	118	123	228	233	238	243	248	353	358	363	368	373	478	483	488	493	498
104	109	114	119	124	229	234	239	244	249	354	359	364	369	374	479	484	489	494	499
105	110	115	120	125	230	235	240	245	250	355	360	365	370	375	480	485	490	495	500

Name ____

Address ____

City ______________________ State________ Zip________

Kilobaud Microcomputing • September 1981

Microcomputing • POB 2741 • Clinton IA 52735

BOOKS

Please send me the following Microcomputing products:

Qty.	Catalog #	Title	Unit Price	Total

Add $1 shipping/handling________

Total________

Please allow 4–6 weeks for delivery. No C.O.D.s accepted.

Enclosed $____________ ☐ Check ☐ M.O.

Bill: ☐ AE ☐ MC ☐ Visa

Card # ______________________________ Exp. date________

Signature ____________________ Interbank # ____________

Name ____

Address ____

City ____

State ______________________________ Zip__________

9-81

Microcomputing • Mail Order Dept. • Peterborough NH 03458

SUBSCRIPTION

MICROCOMPUTING subscribers save $10 off the newsstand price.

- ☐ New subscription ☐ Renewal
- ☐ 1 year —$25
- ☐ 2 years—$38
- ☐ 3 years—$53

Enclosed $____________ ☐ Check ☐ M.O.

Bill: ☐ MC ☐ Visa ☐ AE ☐ me

Card # ______________________________ Exp. date________

Signature ____________________ Interbank # ____________

Name ____

Address ____

City ______________________ State______________ Zip________

Canada—$27 1 year only US funds

Other foreign—$35 1 year only US funds

319B9

Microcomputing • POB 997 • Farmingdale NY 11737

kilobaud MICROCOMPUTING

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 217 CLINTON IA 52735

kilobaud POSTAGE WILL BE PAID BY ADDRESSEE
MICROCOMPUTING T.M.
POB 2741
Clinton IA 52735

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO 1024 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE
kilobaud
MICROCOMPUTING T.M
Subscription Dept.
POB 997
Farmingdale NY 11737

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO 1024 PETERBOROUGH NH 03458

POSTAGE WILL BE PAID BY ADDRESSEE
kilobaud
MICROCOMPUTING T.M.
Peterborough NH 03458

Att. Mail Order

LETTERS

(from page 33)

either of these programs (and much much more) are available from Lazy Writer. Not only does this program offer superb word processing capability and an easy to follow manual, but it includes a very useful communications package. The epitome of superb service is the release of free updates of the software. Microsoft did that with their BASIC Compiler. When I purchased Lazy Writer (version 1.7), there were promises of better things to come. It is hard to believe but the updated, enhanced version (1.8) actually delivers on those promises, and the update was free (it even arrived on its own diskette). Anyone looking for a word processor on the TRS-80 should seriously compare any other program to Lazy Writer.

Now for the bad news. Nearly a year ago I purchased a Selectric typewriter interface from Escon Products (1219 Alcosta Blvd., San Ramon, CA 94583, 415-820-1256). I purchased this interface mostly on the basis of the favorable review printed in *Microcomputing* (March 1980, p. 120).

I never could get the interface to work except in the test mode, even after returning the interface three times. I finally returned the entire interface in February of this year for a refund (they said I could). They received it on Feb. 5, but I have yet to get an answer to any of my letters or have any of my phone calls returned. It seems that both Mr. Shaw and the accountant have always left or stepped out. I have not received a refund and I am beginning to wonder if I have lost my $600. Not only do I not have the $600, but I don't even have the interface (not that it would do me any good)!

Konrad Kuzmanoff
Columbus, OH

On Names and VDTs

Your "Name Games" (As the Word Turns, June 1981) hit home. To my everlasting regret, I encumbered my own firm with the name Sociomatrix. Every use of it over the phone requires that it be spelled. Worse yet, it is lost in the sea of similar such names as you listed in your column. Sociomatrix is forgettable. But who can forget Pickles & Trout?

Your thorough article on the physiological effects of working with CRTs has already been put to use. I am drawing up specifications for a computer system to be purchased by a client. Before reading your article, I had scarcely given a thought to the human factors involved in using terminals. Now, I will take into account the arrangement of work places and the design of the CRTs. Thanks.

William James Haga
Monterey, CA

Second Scramble

The Scramble article in the January 1981 issue ("Scramble" by Edward Rager, p. 78) was interesting. I proceeded to develop a general algorithm for scrambling an input string of any length. The algorithm makes use of the recursive nature of scrambling or permuting characters and uses equated FOR-NEXT loops to generalize the process. The algorithm works by taking all permutations of the first two, three, . . . , L leftmost characters in the input (I$) string and storing them in appropriate string array elements (A$ (*)) for subsequent processing. When the permutations of all characters in the string except the rightmost (Len (I$) − 1) have been established, then the stored strings are used as seeds for direct processing and screen display.

Permutations are developed by character concatenation and right rotation of input string substrings. For example, suppose ABC is the input string using substring AB and right rotating once gives BA. BA and AB constitute the two permutations consisting of any two characters. (Replace A by first character, B by second character). Continue by concatenating the third character C onto the two previous permutations AB and BA. Now right rotate ABC and BAC each two times to get ABC, CAB, BCA, and BAC, CBA, ACB. By simple repetition, the permutations of any length of string could be formed. Naturally, a computer lends itself to these operations quite nicely.

The program in Listing 2 will develop permutations in the described manner and output them in a page format for easy viewing. The program is quite short (approx. 1K) and, although written for Commodore BASIC 4.0, it should be adaptable to any BASIC. Since the program needs working array A$ dimensioned to (Len (I$) − 1)!, the amount of available RAM will determine the largest possible input string. A PET having 32K allows input strings of seven characters ($6 \times (1 \times 2 \times 3 \times 4 \times 5 \times 6) = 4320$ bytes) while larger strings cause dimensioning errors to occur. (Eight characters $7 \times 7! = 7 \times 5040 = 35280$ bytes.)

Having demonstrated a short, more generalized approach to this problem, I would like to say that Scramble was a poorly thought-out program. Granted, it did use subroutines extensively, but their usage was not dictated by the circumstances. In the future, I would like to see more clever programs in your magazine.

Bill Theisen
Madison, WI

Book Rook

I would just like to concur with Brian Klinger's review of *Mostly BASIC: Applications for Your PET*. He did omit one point, which I found particularly annoying after having invested $10.95.

The same program is replicated in the book four times as four different programs. The Foreign Language Flash Card program appears for the French, Spanish, Italian and German languages. Each of the programs occupies three pages, with the only differences being seven DATA statement lines consisting of 25 foreign language words and their respective translations, and one or two prompt statements in the program. It seems that this is nothing more than an attempt to increase the volume of the book. It would have been sufficient to list

```
10 REM            WORD PERMUTATION PROGRAM
20 REM                  BILL THEISEN
30 REM                  JUNE 10,1981
40 REM
50 PRINT"{CLR}":CLR:REM                                  CLEAR SCREEN/ARRAY
60 INPUT"CHARACTERS TO BE SCRAMBLED";I$
70 L=LEN(I$):IFL<1 OR L>7 GOTO 60:REM                    TEST STRING LENGTH
80 F=1:FOR A=1 TO L-1:F=F*A:NEXTA:REM                    CALCULATE (L-1)!
90 DIMA$(F):K=1:REM                                      INITIALIZE ARRAY/VARIABLE
100 A$(0)=LEFT$(I$,1):REM                                ENTER 1ST CHAR IN A$
110 FORC=1 TO L-1 STEP 1:K=K*C  :REM                     LOOP1
120 FOR B=0 TO F-(F/K) STEP F/K:REM                      LOOP2
130 A$(B)=A$(B)+MID$(I$,C+1,1):S$=A$(B):REM              CONCATENATE NEW CHAR
140 FOR A=F/(K*(C+1)) TO F*C/(K*(C+1)) STEP F/(K*(C+1)):REM LOOP3
150 S$=RIGHT$(S$,1)+LEFT$(S$,LEN(S$)-1):A$(A+B)=S$:REM ROTATE RIGHT
160 NEXTA:NEXTB:NEXTC
170 REM*****      PAGE PRINT ROUTINE      *****
180 P=1:PRINT"{CLR}PAGE ";P:PRINT
190 FORI=0 TO F-1:S$=A$(I):REM                           INITIALIZE SHIFT STRING
200 FOR J=0 TO L-1:PRINTS$,:REM                          ROTATE LOOP
210 S$=RIGHT$(S$,1)+LEFT$(S$,LEN(S$)-1):REM              ROTATE RIGHT
220 NEXTJ:PRINT
230 IFINT((I+1)/20)<>(I+1)/20  THEN 270:REM              PAGE FULL?
240 PRINT:PRINT"HIT ANY KEY TO CONTINUE":REM             NEXT PAGE?
250 GETI$:IFI$<>""THEN P=P+1:PRINT"{CLR}PAGE ";P:PRINT:GOTO270
260 GOTO250
270 NEXTI
280 PRINT:PRINT"HIT ANY KEY TO RERUN":REM                 RERUN?
290 GETI$:IFI$<>"" THEN 50
300 GOTO290
```

Listing 2.

the program one time and indicate what changes would be necessary in order to modify it for a different language.

I am surprised that a reputable publisher like Howard W. Sams & Company would allow something like this to slip through their editorial process.

I would also like to take this opportunity to mention that I look forward to your fine magazine arriving each month. Keep up the excellent work.

Stanley M. Berlin
Dallas, TX

You Can't Get There from Here

I was shocked to see J. C. Hassall's multipurpose power supply design in your July 1981 *Microcomputing* ("Construct a Modular, Multi-Purpose Power Supply," p. 61).

Apparently your editors do not screen the articles for accuracy. I thought any novice designer knows you can't get regulated 5 V from a 6.3 V filament transformer at anywhere near rated current. If everything is 100 percent efficient:

6.3 V RMS $\times$ 1.414 = 8.91 V Peak
8.91 V $-$ 2(.7) diode drops = 7.51 V

A 7805 regulator needs at least 7.0 volts, so ripple in the filter must be .51 V or less.

The formula for capacitor selection is:

$$C = \frac{\text{I Load}}{\text{V Ripple}} \times 6 \times 10^{-3}$$

Thus,

$$C = \frac{1.5 \times 6 \times 10^{-3}}{.51} = 17{,}000\ \mu\text{F}$$

The filter cap would have to be at least 17,000 μF for this supply to regulate properly.

Also, design rules say transformer current rating for a fullwave bridge with capacitive input filter should be 1.8 times the dc output current. So, a transformer with 2.7 ARMS should be used for a 1.5 A supply. Mr. Hassall's supply may work on light loads but will not work anywhere near 1.5 A loads, especially with a 1.2 A transformer.

Robert L. Ribbeck
Lackawanna, NY

Response:

Mr. Ribbeck is fundamentally correct. However, his premise is that a filament transformer is used for the +5 V dc supply. To realize full current from the supply, a transformer rated at least for 3 A should be used. I am guilty of a sin of omission, which may account for the transformer confusion. I asked my local Radio Shack store for transformer part numbers but did not personally verify them. The R-S part number 273-050 transformer will not work in the design, and I am indebted to Mr. Ribbeck for catching the error.

Mr. Ribbeck's comments about the filter capacitor values are well-taken. He has shown that the +5 V filter capacitor should be 17,000 μF or larger for proper regulation at 1.5 A. However, at 2 A output current with the 4,000 μF filter capacitor, measured output ripple was less than 0.15 V. By paralleling a 50,000 μF capacitor, the ripple was reduced less than 0.05 V. That much ripple would be unacceptable for a laboratory power supply, for example, but my first homebrew 8080 based system ran for over two years using the published circuit.

Interested readers who wish to learn more about power supply design are referred to William Cattey's article "Heavy Duty Power Supply" in the April 1977 issue of *Microcomputing* (p. 78), National Semiconductor's *Voltage Regulator Handbook* or American Radio Relay League's *Radio Amateur's Handbook*.

J. C. Hassall
Blacksburg, VA

Test Your Readability

My congratulations to you and Richard R. Parry for the outstanding article, "Minding Your P's and Q's" in the June issue (p. 58). I especially found his discussion of the sentence and word algorithms very useful.

I am a firm believer in readability testing and use it extensively with articles I write for several publications.

Please let your readers know that there is a fast assembly-language readability testing program available for any 6800 or 6809 microprocessor running the Flex operating system. It uses theories promoted by Dr. Rudolf Flesch in *The Art of Plain Talk*.

I wrote READTEST about a year ago and it is being marketed by Frank Hogg Laboratory, 130 Midtown Plaza, 700 East Water St., Syracuse, NY 13210. READTEST sells for $54.95 and comes with an education manual that will be an immense help to any writer. The source code may also be purchased for an additional $20.

Dale L. Puckett
Woodbridge, VA

More on Paper Tiger

A copy of Jim Hansen's fine article about the Paper Tiger, "A Tiger's-Eye View of Computer Graphics," May 1981, p. 184, was sent to me by an Ohio reader. I'd like to refer Mr. Hansen and your readers to the February 1981 issue of *80 Microcomputing* in which I describe and give a listing for a screenprinter for the Tiger (IDS 440). The program is written in BASIC for the TRS-80 Model I. The May issue of *80 Microcomputing* on page 16 gives an extension to my program, submitted by William D. Webb, to include lowercase characters (remove the remark statements, put 'NEXT' on line 10090 on a separate line).

Ruth Lewart
Holmdel, NJ

Jim Hansen's piece was the most useful micro article I've read in a year. By explaining how the TRS-80 eats and transforms control codes, it saved me an unnecessary trip to the repair shop. Thank you, and please encourage Mr. Hansen to publish more on Tiger graphics.

W. R. Nugent
Reston, VA

Jim has several projects in the works for future issues.—Eds.

MICRO QUIZ

(from page 23)

Answer: -121

The final value of z is the sum of its initial value and the contributions from each of the if statements.

The first if statement contributes $-0, -3, -6, -9, \ldots, -24 = -3*(1+2+\ldots+8) = -3*(8*9/2)$ to the sum.

The second if statement contributes -2 for all nonnegative integers less than or equal to 25, which are not multiples of 4. Since there are 7 multiples of 4 (0, 4, . . . , 24), this statement contributes $-2*(26-7)$ to the sum.

Thus, the final value of z is $25 - 108 - 38$.

OSI

AARDVARK NOW MEANS BUSINESS!

OSI

WORD PROCESSING THE EASY WAY— WITH MAXI-PROS

This is a line-oriented word processor designed for the office that doesn't want to send every new girl out for training in how to type a letter.

It has automatic right and left margin justification and lets you vary the width and margins during printing. It has automatic pagination and automatic page numbering. It will print any text single, double or triple spaced and has text centering commands. It will make any number of multiple copies or chain files together to print an entire disk of data at one time.

MAXI-PROS has both global and line edit capability and the polled keyboard versions contain a corrected keyboard routine that make the OSI keyboard decode as a standard typewriter keyboard.

MAXI-PROS also has sophisticated file capabilities. It can access a file for names and addresses, stop for inputs, and print form letters. It has file merging capabilities so that it can store and combine paragraphs and pages in any order.

Best of all, it is in BASIC (0S65D 51/4" or 8" disk) so that it can be easily adapted to any printer or printing job and so that it can be sold for a measly price.

MAXI-PROS — $39.95

NEW-NEW—NEW TINY COMPILER

The easy way to speed in your programs. The tiny compiler lets you write and debug your program in Basic and then automatically compiles a Machine Code version that runs from 50-150 times faster. The tiny compiler generates relocatable, native, transportable machine code that can be run on any 6502 system.

It does have some limitations. It is memory hungry — 8K is the minimum sized system that can run the Compiler. It also handles only a limited subset of Basic — about 20 keywords including FOR, NEXT, IF THEN, GOSUB, GOTO, RETURN, END, STOP, USR(X), PEEK, POKE, -,=,*,/, (,),<> ,Variable names A-Z, and Integer Numbers from 0-64K.

TINY COMPILER is written in Basic. It can be modified and augmented by the user. It comes with a 20 page manual.

TINY COMPILER — $19.95 on tape or disk

THE AARDVARK JOURNAL

FOR OSI USERS — This is a bi-monthly tutorial journal running only articles about OSI systems. Every issue contains programs customized for OSI, tutorials on how to use and modify the system, and reviews of OSI related products. In the last two years we have run articles like these!

1) A tutorial on Machine Code for BASIC programmers.
2) Complete listings of two word processors for BASIC IN ROM machines.
3) Moving the Directory off track 12.
4) Listings for 20 game programs for the OSI.
5) How to write high speed BASIC — and lots more —

Vol. 1 (1980) 6 back issues - $9.00
Vol. 2 (1981) 2 back issues and subscription for 4 additional issues - $9.00.

ACCOUNTS RECEIVABLE — This program will handle up to 420 open accounts. It will age accounts, print invoices (including payment reminders) and give account totals. It can add automatic interest charges and warnings on late accounts, and can automatically provide and calculate volume discounts.

24K and 0S65D required, dual disks recommended. Specify system.
Accounts Receivable. $99.95

* * * SPECIAL DEAL — NO LESS! * * *

A complete business package for OSI small systems — (C1, C2, C4 or C8). Includes MAXI-PROS, GENERAL LEDGER, INVENTORY, PAYROLL AND ACCOUNTS RECEIVABLE — ALL THE PROGRAMS THE SMALL BUSINESS MAN NEEDS. $299.95

P.S. We're so confident of the quality of these programs that the documentation contains the programmer's home phone number!

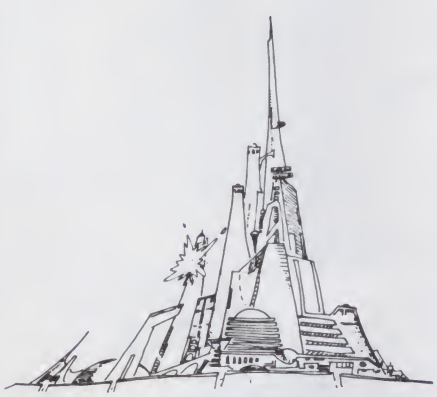
SUPERDISK II

This disk contains a new BEXEC* that boots up with a numbered directory and which allows creation, deletion and renaming of files without calling other programs. It also contains a slight modification to BASIC to allow 14 character file names.

The disk contains a disk manager that contains a disk packer, a hex/dec calculator and several other utilities.

It also has a full screen editor (in machine code on C2P/C4)) that makes corrections a snap. We'll also toss in renumbering and program search programs — and sell the whole thing for — SUPERDISK II $29.95 (5 1/4") $34.95 (8").

AND FUN, TOO!



BOOKKEEPING THE EASY WAY — WITH BUSINESS I

Our business package 1 is a set of programs designed for the small businessman who does not have and does not need a full time accountant on his payroll.

This package is built around a **GENERAL LEDGER** program which records all transactions and which provides monthly, quarterly, annual, and year-to-date PROFIT AND LOSS statements. GENERAL LEDGER also provides for cash account balancing, provides a BALANCE SHEET and has modules for DEPRECIATION and LOAN ACCOUNT computation.
GENERAL LEDGER (and MODULES) $129.95.

PAYROLL is designed to interface with the GENERAL LEDGER. It will handle annual records on 30 employees with as many as 6 deductions per employee.
PAYROLL - $49.95.

INVENTORY is also designed to interface with the general ledger. This one will provide instant information on suppliers, initial cost and current value of your inventory. It also keeps track of the order points and date of last shipment.
INVENTORY - $59.95.

GAMES FOR ALL SYSTEMS

GALAXIAN - 4K - One of the fastest and finest arcade games ever written for the OSI, this one features rows of hard-hitting evasive dogfighting aliens thirsty for your blood. For those who loved (and tired of) Alien Invaders. Specify system — A bargain at $9.95

NEW — NEW — NEW

LABYRINTH - 8K - This has a display background similar to MINOS as the action takes place in a realistic maze seen from ground level. This is, however, a real time monster hunt as you track down and shoot mobile monsters on foot. Checking out and testing this one was the most fun I've had in years! — $13.95.

NIGHT RIDER - You've seen similar games in the arcades. You see a winding twisting road ahead as you try to make time and stay on the road. NIGHT RIDER uses machine code to generate excellent high speed graphics - by the same author as MINOS.
NIGHT RIDER — $12.95 cassette only

THIEF - Another machine code goody for the C1P cassette only. You must use mobile cannon to protect the valuable jewels in the middle of the screen from increasingly nasty and trigger happy thiefs. Fast action and fun for one or two players. THIEF $13.95 on C1 cassette only!

SUPPORT ROMS FOR BASIC IN ROM MACHINES — C1S/C2S. This ROM adds line edit functions, software selectable scroll windows, bell support, choice of OSI or standard keyboard routines, two callable screen clears, and software support for 32-64 characters per line video. Has one character command to switch model 2 C1P from 24 to 48 character line. When installed in C2 or C4 (C2S) requires installation of additional chip. C1P requires only a jumper change. — $39.95
C1E/C2E similar to above but with extended machine code monitor. — $59.95

Please specify system on all orders

This is only a partial listing of what we have to offer. We now offer over 100 programs, data sheets, ROMS, and boards for OSI systems. Our $1.00 catalog lists it all and contains free program listings and programming hints to boot.

VISA

master charge

AARDVARK TECHNICAL SERVICES, LTD. ✓91
2352 S. Commerce, Walled Lake, MI 48088
(313) 669-3110

OSI OSI

✓See List of Advertisers on page 210

Edited by Linda Stephenson

80K Stratos System
Typewriter/Printer
MiniFrame from MicroDaSys
68,000 Micros

Turnkey Microcomputer System

The Stratos single-board Z-80 microcomputer comes equipped with CP/M 2.2, Spellbinder word processing and office management software, a full financial package, CBASIC or Pascal compiler and a wide range of utility programs unique to the Stratos. The Font utility lets the user create, store and recall any set of up to 192 characters, allowing keystroke generation and display of foreign alphabets, mathematical symbols or the user's own special characters. The Reminder program creates and accesses a 99-year calendar file of user-entered dates, times and messages or functions. The Reminder will automatically perform the selected task at the chosen time. Autophone is an auto-originate/auto-answer repeat-dialing telephone control program; it supports a 50-name directory, selectable baud rates and seven- or eight-bit operations.

The system is delivered with 80K RAM and 1.2M or .65M of on-line storage. Teak cabinets enclose the system and drives and the separate keyboard with 19 special-function keys. The fully decodeable port bus allows the Stratos to function as a member or controller of a distributed intelligence network, performing all of its standard functions in local mode and accessing a common database and shared peripherals through the network. Prices vary with configuration.

The Stratos 80K microcomputer from Symbiotic Systems, Inc.

Symbiotic Systems, Inc., 118 Naglee Ave., Santa Cruz, CA 95060. Reader Service number 498.

Electronic Typewriter/Printer

The ES 100 RO is both an electronic typewriter and a computer printer. Its built-in RS-232C serial interface lets you transform it into a computer printer compatible with most standard microcomputers. The ES 100 RO has a patented interchangeable printwheel system which prints up to 17 characters per second, pica or elite. The typewriter has one-step correction and an eight-character memory that lifts off a word or character neatly and easily. Electronic typing is faster and more efficient, and having fewer moving parts means less servicing. The ES 100 RO is priced at $1690.

Olympia USA, Box 22, Somerville, NJ 08876. Reader Service number 496.

The Olympia ES 100 RO electronic typewriter/printer.

Dual Processor Computer

The 68K MiniFrame lets you tailor your system to your computing needs. Systems that can be configured around the MiniFrame range from a single-user system that emulates CP/M software to a full-powered multi-user system with 16M LRU demand paged virtual memory, 2M of RAM and the UNIX operating system. Relocation and limit registers are maintained for each user to support multi-user and multitasking operations. It is designed to go at a full 12 MHz, and up to four gigabytes can be addressed.

The basic unit uses a 68,000 as its main processor and a 6809 for memory management, I/O and housekeeping operations. The price is $8600.

MicroDaSys, Inc., PO Box 36215, Los Angeles, CA 90036. Reader Service number 481.

MicroDaSys' 68K MiniFrame computer.

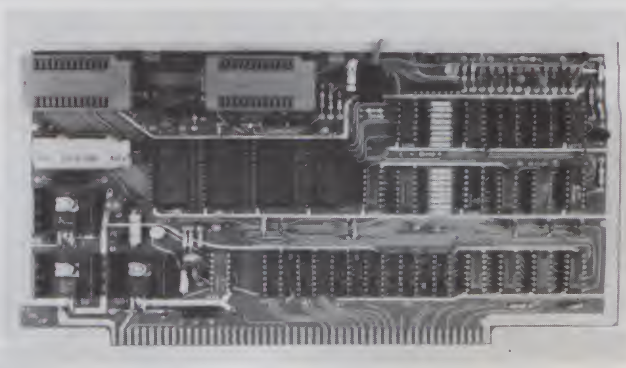
The Eagle 32 Model 20 microcomputer from Computhink.

Two New 68,000 Micros

Computhink, 965 West Maude Ave., Sunnyvale, CA 94086, has introduced the Eagle 32 line of microcomputer systems. Both systems are based on the Motorola 68,000 16/32-bit microprocessor chip. Model 10 has 1.6M storage capacity expandable to 3.2M on a 5-1/4-inch floppy disk; the Model 20 has 2.4M, expandable to 4.8M. Both systems have a 128K main memory, expandable to 1M, and an 80×24 display with reverse video and flashing. The Eagle 32 is primarily a dedicated commercial or scientific application system; it includes a batch operating system, BASIC+, FORTH+ and Tiny FORTRAN. The Eagle 32 Model 10 costs $6650; the Model 20 costs $7650. Reader Service number 486.

S-100 Board Combines Programming and EPROM Memory

The PB1 dual-service S-100 board has both programming and memory functions, thus improving use of slot space in S-100-based computers. Four on-card sockets provide either 4096 or 8192 bytes of memory, depending upon whether 2708 or 2716 EPROMs are used. Two programming sockets are also provided, one for a 2708 and one for a 2716—both are DIP-switch programmable to any 4K boundary. The use of Textool sockets would mean that flipping a lever would lift an EPROM out of a programming socket, thus preventing damage to both the EPROM and the board itself. Separate on-card circuits allow programming of 2708 and 2716 (5 V) EPROMs without board modifications. The programming voltage is generated on-board, eliminating any need for an external supply. Unused EPROM sockets do not occupy memory space, so the plug-in memory feature assures that the user is not committed to a full 4K or 8K memory complement. The PB1 board, with needed software, is priced at $265, assembled and tested; a kit version is $179.



SSM Microcomputer Products' PB1 single-unit programming and memory board.

✓See List of Advertisers on page 210

The New World Computer Company's fixed and removable cartridge drive.

SSM Microcomputer Products, 2190 Paragon Drive, San Jose, CA 95131. Reader Service number 494.

Removable Cartridge Drive

The New World Computer Company, 3176 Pullman St., #120, Costa Mesa, CA 92626, has introduced a 5-1/4-inch fixed and removable cartridge drive. The removable cartridge adds Winchester capacity while maintaining bubble memory performance of the company's low-mass multiple-head drives. Mikro-Disc V drives are available in five models, ranging from the Model 2/0 with 2M fixed storage, priced at under $500, to the Model 4/4, with 4M fixed and 4M removable storage and priced under $1200. Two interface versions are available, for compatibility with New World Mikro-Disc drives and for Seagate or Shugart drives. Reader Service number 482.

Cromemco-Compatible RAM Board

Now there is a fast, reliable 64K memory board that is compatible with any S-100 bus-based Cromemco CROMIX system. The SUPERAM 4C is organized into two blocks of 32K, each of which can be placed in any of eight different memory banks. Bank selection is available on 32K boundaries. Address and bank assignment of each 32K block is selectable via simple switch settings. SUPERAM 4C features extended bank select for up to 16M expansion. Memory refresh is transparent to the processor. Memory access time is 250 ns. Operating speed is 4 MHz with no wait states. It requires +8 V at 0.8 A, and +18 V at 0.2 A. Price is $995.

The SUPERAM 4C memory board from Piiceon.

NEW! TPM* for TRS-80 Model II
NEW! System/6 Package
Computer Design Labs

Z80* Disk Software

We have acquired the rights to all TDL software (& hardware). TDL software has long had the reputation of being the best in the industry. Computer Design Labs will continue to maintain, evolve and add to this superior line of quality software.
— Carl Galletti and Roger Amidon, owners.

Software with Manual/Manual Alone

All of the software below is available on any of the following media for operation with a Z80 CPU using the CP/M* or similar type disk operating system (such as our own TPM*).

for TRS-80* CP/M (Model I or II)
for 8" CP/M (soft sectored single density)
for 5¼" CP/M (soft sectored single density)
for 5¼" North Star CP/M (single density)
for 5¼" North Star CP/M (double density)

BASIC I

A powerful and fast Z80 Basic interpreter with EDIT, RENUMBER, TRACE, PRINT USING, assembly language subroutine CALL, LOADGO for "chaining", COPY to move text, EXCHANGE, KILL, LINE INPUT, error intercept, sequential file handling in both ASCII and binary formats, and much, much more. It runs in a little over 12 K. An excellent choice for games since the precision was limited to 7 digits in order to make it one of the fastest around. $49.95/$15.

BASIC II

Basic I but with 12 digit precision to make its power available to the business world with only a slight sacrifice in speed. Still runs faster than most other Basics (even those with much less precision). $99.95/$15.

NEW MANUAL

BUSINESS BASIC

The most powerful Basic for business applications. It adds to Basic II with random or sequential disk files in either fixed or variable record lengths, simultaneous access to multiple disk files, PRIVACY command to prohibit user access to source code, global editing, added math functions, and disk file maintenance capability without leaving Basic (list, rename, or delete). $179.95/$25.

ZEDIT

A character oriented text editor with 26 commands and "macro" capability for stringing multiple commands together. Included are a complete array of character move, add, delete, and display function. $49.95./$15.

ZTEL

Z80 Text Editing Language - Not just a text editor. Actually a language which allows you to edit text and also write, save, and recall programs which manipulate text. Commands include conditional branching, subroutine calls, iteration, block move, expression evaluation, and much more. Contains 36 value registers and 10 text registers. Be creative! Manipulate text with commands you write using Ztel. $79.95/$25.

TOP

A Z80 Text Output Processor which will do text formatting for manuals, documents, and other word processing jobs. Works with any text editor. Does justification, page numbering and headings, spacing, centering, and much more! $79.95/$25.

MACRO I

A macro assembler which will generate relocateable or absolute code for the 8080 or Z80 using standard Intel mnemonics plus TDL/Z80 extensions. Functions include 14 conditionals, 16 listing controls, 54 pseudo-ops, 11 arithmetic/logical operations, local and global symbols, chaining files, linking capability with optional linker, and recursive/reiterative macros. This assembler is so powerful you'll think it is doing all the work for you. It actually makes assembly language programming much less of an effort and more creative. $79.95/$20.

MACRO II

Expands upon Macro I's linking capability (which is useful but somewhat limited) thereby being able to take full advantage of the optional Linker. Also a time and date function has been added and the listing capability improved. $99.95/$25.

LINKER

How many times have you written the same subroutine in each new program? Top notch professional programmers compile a library of these subroutines and use a Linker to tie them together at assembly time. Development time is thus drastically reduced and becomes comparable to writing in a high level language but with all the speed of assembly language. So, get the new CDL Linker and start writing programs in a fraction of the time it took before. Linker is compatible with Macro I & II as well as TDL/Xitan assemblers version 2.0 or later. $79.95/$20.

DEBUG I

Many programmers give up on writing in assembly language even though they know their programs would be faster and more powerful. To them assembly language seems difficult to understand and follow, as well as being a nightmare to debug. Well, not with proper tools like Debug I. With Debug I you can easily follow the flow of any Z80 or 8080 program. Trace the program one step at a time or 10 steps or whatever you like. At each step you will be able to see the instruction executed and what it did. If desired, modifications can then be made before continuing. It's all under your control. You can even skip displaying a subroutine call and up to seven breakpoints can be set during execution. Use of Debug I can pay for itself many times over by saving you valuable debugging time. $79.95/$20.

DEBUG II

This is an expanded debugger which has all of the features of Debug I plus many more. You can "trap" (i.e. trace a program until a set of register, flag, and/or memory conditions occur). Also, instructions may be entered and executed immediately. This makes it easy to learn new instructions by examining registers/memory before and after. And a RADIX function allows changing between ASCII, binary, decimal, hex, octal, signed decimal, or split octal. All these features and more add up to give you a very powerful development tool. Both Debug I and II must run on a Z80 but will debug both Z80 and 8080 code. $99.95/$20.

ZAPPLE

A Z80 executive and debug monitor. Capable of search, ASCII put and display, read and write to I/O ports, hex math, breakpoint, execute, move, fill, display, read and write in Intel or binary format tape, and more! on disk

APPLE

8080 version of Zapple

NEW

NEW! TPM now available for TRS-80 Model II!

TPM*

A NEW Z80 disk operation system! This is not CP/M*. It's better! You can still run any program which runs with CP/M* but unlike CP/M* this operating system was written specifically for the Z80* and takes full advantage of its extra powerful instruction set. In other words its not warmed over 8080 code! Available for TRS-80* (Model I or II). Tarbell, Xitan DDDC, SD Sales "VERSA-FLOPPY", North Star (SD&DD), and Digital (Micro) Systems. $79.95/$25.

SYSTEM MONITOR BOARD (SMB II)

A complete I/O board for S-100 systems. 2 serial ports, 2 parallel ports, 1200/2400 baud cassette tape interface, sockets for 2K of RAM, 3-2708/2716 EPROM's or ROM, jump on reset circuitry. Bare board $49.95/$20.

ROM FOR SMB II

2KX8 masked ROM of Zapple monitor. Includes source listing $34.95/$15.

PAYROLL (source code only)

The Osborne package. Requires C Basic 2.
5" disks $124.95 (manual not included)
8" disks $ 99.95 (manual not included)
Manual $20.00

ACCOUNTS PAYABLE/RECEIVABLE (source code only)

By Osborne, Requires C Basic 2
5" disks $124.95 (manual not included)
8" $99.95 (manual not included)
Manual $20.00

GENERAL LEDGER (source code only)

By Osborne. Requires C Basic 2
5" disks $99.95 (manual not included)
8" disks $99.95 (manual not included)
Manual $20.00

C BASIC 2

Required for Osborne software. $99.95/$20.

SYSTEM/6

NEW

TPM with utilities, Basic I interpreter, Basic E compiler, Macro I assembler, Debug I debugger, and ZEDIT text editor.
Above purchased separately costs $339.75
Special introductory offer: Only $179.75 with coupon!!

$160.

This Coupon Is Worth One Hundred And Sixty Dollars Toward The Full Price Of The SYSTEM/6 Package
System/6 with this coupon is only $179.95. This is a limited time offer.

$160.00

ORDERING INFORMATION

Visa, Master Charge and C.O.D. O.K. To order call or write with the following information.

1. Name of Product (e.g. Macro I)
2. Media (e.g. 8" CP/M)
3. Price and method of payment (e.g. C.O.D.) include credit card info. if applicable.
4. Name, Address and Phone number.
5. For TPM orders only: Indicate if for TRS 80, Tarbell, Xitan DDDC, SD Sales (5¼" or 8"). ICOM (5¼" or 8"), North Star (single or double density) or Digital (Micro) Systems.
6. N.J. residents add 5% sales tax.

Manual cost applicable against price of subsequent software purchase in any item except for the Osborne software.

For information and tech queries call
609-599-2146

For phone orders ONLY call toll free
1-800-327-9191
Ext. 676
(Except Florida)

OEMS

Many CDL products are available for licensing to OEMs. Write to Carl Galletti with your requirements.

* Z80 is a trademark of Zilog
* TRS-80 is a trademark for Radio Shack
* TPM is a trademark of Computer Design Labs. It is not CP/M*
* CP/M is a trademark of Digital Research

Prices and specifications subject to change without notice.

DEALER INQUIRIES INVITED.

COMPUTER DESIGN LABS

✓18

342 Columbus Avenue
Trenton, N.J. 08629

✓See List of Advertisers on page 210

The Series 5000 speech data entry system from Heuristics.

Heuristics' Series 7000 Voice Controller System.

Piiceon, 2350 Bering Drive, San Jose, CA 95131. Reader Service number 492.

Speech Recognition Products

Two new stand-alone voice recognition terminals permit rapid installation of speech recognition by eliminating all system integration. There is no hidden cost of software development because local storage allows voice data entry to be transparent to the host computer. Heuristics' proprietary algorithms transform this data into a compact characteristic reference template. The template, obtained during training, is compared during recognition with the audio input. When a match occurs, a user-defined ASCII character string is sent to the host and/or the terminal.

Series 5000 products are speech data entry devices installed inside Lear Siegler ADM 3 and 5 terminals, forming complete self-contained voice/data entry terminals; the system costs $4995. The Series 7000 is an intelligent speech data entry device used in conjunction with any ASCII terminal; it costs $4595.

Heuristics, 1285 Hammerwood Ave., Sunnyvale, CA 94086. Reader Service number 491.

TRS-80 Direct-Connect Modem

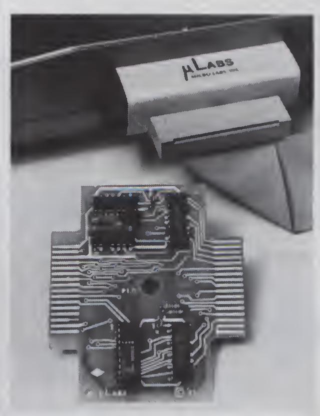
A low-cost modem available from Tandy Corporation/Radio Shack, 1800 One Tandy Center, Fort Worth, TX 76102, expands the possible uses for your TRS-80. This plug-in device enables you to access information networks and communicate with other computers over the telephone. The direct-connect technique eliminates the data losses and errors which can be caused by microphonics in an acoustic coupler. The TRS-80 Modem I permits direct electrical connection to standard telephone lines for up to 300 baud full duplex (simultaneous two-way) communications when connected to the computer's RS-232 port; Modem I provides half-duplex operation (one-way at a time) for 16K Level II TRS-80 Model I systems without RS-232, through the computer's cassette port. Power is supplied through an ac power module. The modem unit costs $149. Reader Service number 484.

Radio Shack's new low-cost TRS-80 Modem I.

Color Printer Interface

The CPRINT module gives your TRS-80 Color Computer a plug-compatible Centronics-type parallel printer port for use with all parallel Radio Shack, Centronics, Epson and similar printers. Software contained in permanent onboard memory gives transparent operation with several added features: all LLIST and PRINT #-2 output is automatically rerouted; a screen-print function can be initiated at any time; line width can be set; the graphics in the LPVII can be accessed; page length can be set; and blank lines are inserted between pages. The CPRINT module is a fully-buffered eight-bit I/O port which will interface with any Model I/III products which plug into the printer port—e.g., the Percom speech board. CPRINT module in a plastic case is priced at $49.95.

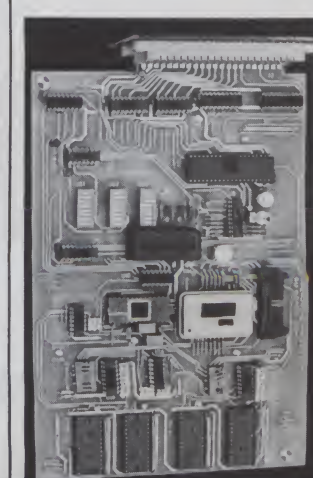
Micro-Labs, Inc., 902 Pinecrest, Richardson, TX 75080. Reader Service number 493.



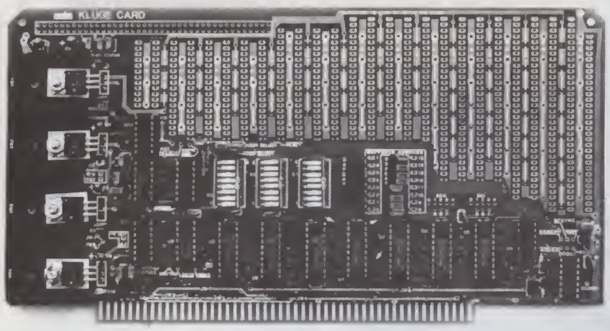
The CPRINT printer interface for the Color Computer, from Micro-Labs, Inc.

AIM-65 Data Logger

The IB-902AB interface card from Columbus Instruments, PO Box 44049, Columbus, OH 43204, transforms the Rockwell AIM-65 computer into a 16-channel analog data acquisition system. It was developed to convert AIM-65 computers into dedicated laboratory and industrial measuring systems. The card contains a fast analog-to-digital converter, 16-channel multiplexer, real-time clock/calendar and memory space for an additional 16K (or up to 64K with new advanced chips). The same card can interface to



Columbus Instruments' IB-902AB interface card.



Ackerman Digital Systems' Kluge Card.

PET, KIM and all 6502 or 6800 systems with a special adapter cable and small modifications. It is supplied with demonstration programs for calibrating and testing purposes. External power supply required. Priced at $1270. Reader Service number 485.

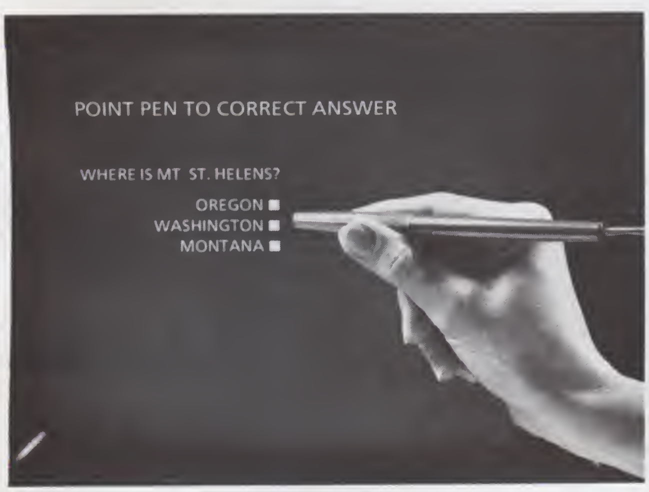
Light Pen for the TRS-80 Model III

A self-contained light pen which plugs directly into the Radio Shack TRS-80 Model III has been announced by the 3G Company, Inc., Rt. 3, Box 28A, Gaston, OR 97119. This pen also works with the TRS-80 Model I. The light pen lets you bypass the keyboard and interact directly with the information displayed on the CRT. This adds versatility to graphics and game programs. It is also useful for children who don't know how to type. The light pen and a demonstration game cassette, sample program and programming booklet cost $39.95. Reader Service number 490.

Two New ADS Products

The ADS PROM Blaster programs most families of EPROMs and is controlled by software running under either CP/M or ADSMON. The PROM Blaster, using a single 28-pin ZIF socket, lets you program 1K/2K/4K, or 8K EPROMs, including three-supply parts. With the high voltage supply on the card, the board functions on the S-100 bus as an I/O device; it has phantom slave and extended device options as well as wait states. The PROM Writer software utility is used to handle all programming functions for all parts supported by the card. Kit price is $199.

Also from ADS is a breadboard with much of the logic built in that you would otherwise wire up every time you build a project. The ADS Kluge Card has bidirectional buffered address and data lines,



The 3G Light Pen simplifies use of educational programs.

See List of Advertisers on page 210

Intertec now offers a cable adapter to connect the SuperBrain microcomputer with its 10M disk storage system.

buffered control lines, on-card wait states, extended device and/or extended memory select. Four on-board power supplies give +5, −5, +12 and −12 V. The bareboard is $39.95; the kit costs $149.95.

Ackerman Digital Systems, Inc., 110 North York Road, Elmhurst, IL 60126. Reader Service number 495.

Ten Megabytes of Storage for SuperBrain

A special adapter cable is available from Intertec Data Systems, 2300 Broad River Road, Columbia, SC 29210, which allows SuperBrain and SuperBrain QD owners to use the CompuStar 10M disk storage system. This adapter connects directly to the processor board of the SuperBrain microcomputer and the rear data port on the DSS. A diskette is supplied for reconfiguring the storage allocations on the disk storage system, and for generating new operating systems to permit communication between the computer and disk storage system. Price is $45. Reader Service number 487.

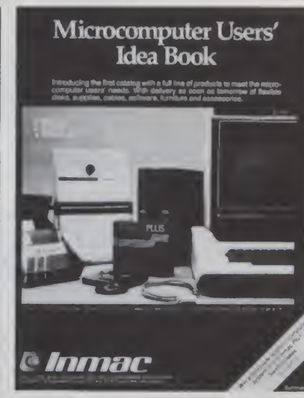
High-Performance Data Concentrator

The Micro800/2 data concentrator, from Micom Systems, Inc., 9551 Irondale Ave., Chatsworth, CA 91311, offers more processing power at a lower price. It permits 16 synchronous or asynchronous data terminals to share a single telephone line. The unit's data compression feature makes it more efficient than most other statistical multiplexers. Automatic retransmission ensures error-free data communications. Concentration rates vary with the computer application, but a factor of 4:1 is usually possible. Price is $1850. Reader Service number 483.

The Micro800/2 data concentrator from Micom Systems, Inc.

Son-of-Inmac Catalog

A free one-year subscription to *The Microcomputer User's Idea Book* can be obtained by writing to Inmac, Department 12, 2465 Augustine Drive, Santa Clara, CA 95051. The 32-page publication lists over 1000 of the latest products—from software packages and CRTs to flexible disks, printer ribbons and EDP-tailored furniture. *The Idea Book* has separate sections listing supply, accessory and cable needs for Apple, Atari, TRS-80 and North Star. Reader Service number 488.



A free catalog with a full line of microcomputer products is being offered by Inmac.

Unlimited Speech Capability

Alford and Associates, PO Box 6743, Richmond, VA 23230, is offering the VS-1 Speaker, a phonetic speech synthesizer for the SS-50 and SS-50C bus. Based on the Votrax SC-01 phoneme synthesizer circuit, the VS-1 provides unlimited speech capability and requires less memory and processor overhead than similar units. The VS-1 occupies a single slot on the SS-50 I/O bus, and is programmed like a parallel printer interface. Ten bytes per second drives the board at its maximum throughput, and about one byte-per-letter in English is required for speech storage. The VS-1 with software and manual costs $229.95. Reader Service number 489.

The VS-1 Speaker phonetic speech synthesizer from Alford and Associates.

Apple Enhancement

Vista Computer Company, 1317 E. Edinger, Santa Ana, CA 92705, is offering its Model 150 Type Ahead Buffer, which is compatible with all Apple II computers and software. Featuring a 40-character type-ahead capability, the Model 150 eliminates the need to wait for computer prompts before entering the next command or data. The Model 150 requires no software patches, cuts or jumpers. Its simplified design affords easy installation. Price is $49.95. Reader Service number 497.

Vista Computer Company's Model 150 Type Ahead Buffer.

See List of Advertisers on page 210

Educational Software
68,000 Interpreter
Dental Insurance Program
Apple III Word Processors

Educational Program

Spelling Bee is designed for children in kindergarten through third grade. The program builds basic spelling skills, develops computer literacy and establishes new vocabulary by linking words and pictures. High-resolution graphics and sound effects help capture the child's interest and reinforce learning. Lesson length and emphasis can be tailored to the individual child's needs. The system requires Applesoft, 48K and DOS 3.3. It costs $29.95.

Edu-Ware Services, Inc., 22222 Sherman Way, Suite 203, Canoga Park, CA 91303. Reader Service number 465.

Games for the Superboard II

Several new game programs are available for your Superboard II microcomputer. They need a minimum of 8K memory, and all joystick games require Aardvark standard joysticks.

Flak Run is a race through flak-filled skies—time and score are continuously updated. Blackjack is the straightforward Vegas version, for up to four players. Breakout is an arcade game that becomes more difficult as you gain skill. Yah-Z is a simple dice game for any number of players. Tank III is a classic tank battle game replete with randomly littered houses, trees, mines, tunnels and barricades. Each player has limited fuel and ammo, and is rewarded for inflicting maximum death and destruction. Cassettes cost $4–$6.

Dee Products, 150 Birchwood, Lake Marion, IL 60110. Reader Service number 473.

APL Interpreter For Motorola 68000

APL.68000 is a full implementation of IBM's APL.SV with the same enhancements used on The Computer Company's commercial timesharing system, ACTION/APL.SV. Included are a powerful file system, fast data search and replacement primitive, special format primitive that allows COBOL-like picture format specifications, overlays, execution of system commands and error trapping. APL.68000 is available for the Motorola VERSAdos operating system.

Micro APL Products Division, The Computer Company/APL Services, 1905 Westmoreland St., Richmond, VA 23230. Reader Service number 478.

Dental Insurance Form Writer

The Dental Insurance Forms program allows rapid billing and claim submission with a minimum of effort. A master form can be created for each patient/family and saved for later use. At future visits the master form is loaded, proposed treatments are entered and the updated form is saved to disk and then printed as a preauthorization form. Later visits require only the loading, editing, saving and printing of the insurance billing. Print as many copies as needed. Each diskette will accommodate over 100 families.

The program requires Apple II with Applesoft, 48K disk drive and printer. Price is $100.

Andent, Inc., 1000 North Ave., Waukegan, IL 60085. Reader Service number 472.

Professional Time Management

Datebook II maintains a complete record of appointments for up to 27 people, allowing quick display of appointment openings and easy access to scheduling information. The operator works from a main option menu. Menu items include scheduling, cancelling, modifying and rescheduling appointments; scheduling a conference; searching for all appointments for a specified person; scanning for openings that satisfy given constraints; inspecting future appointments; and printing a day's appointments. Datebook II is written in Pascal and runs on CP/M, CP/M 86 and UCSD Pascal systems. Price is $295.

Organic Software, 1492 Windsor Way, Livermore, CA 94550. Reader Service number 464.

Oasis Sort Utility

A fast-sort utility for Oasis-based operating systems is available from Quantumn Information Systems, 860 NW 87 St., Seattle, WA 98117. Qisort is a professional utility that operates on both single- and multi-user systems. Qisort will sort ASCII sequential files in ascending or descending order with up to 32 control fields and 255 byte records. Larger records and nonsequential files can be sorted by first doing an extract against that file. Price is $100. Reader Service number 479.

Armchair Baseball

Avalon Hill Game Company, 4517 Harford Road, Baltimore, MD 21214, has released a computer version of its best-selling baseball game. Major League Baseball contains statistics of every major league player for every big league club. As manager, you can effect all the strategies of the real thing. You can re-create an entire season, a championship series and a world series, and get a box score at the end of each game. At the end of your season you get the total season's statistics for every ballplayer. Major League Baseball is available for most popular microcomputers. Cas-

List of Advertisers on page 210

settes cost $25 and disks are $30. Reader Service number 475.

Electronic Spreadsheet For the 6809

Tabula Rasa provides an electronic spreadsheet for 56K 6809 systems running the TSC Extended BASIC interpreter and TSC Macro Assembler. It will operate with most serial terminals and memory-mapped video displays providing at least 16 rows and 64 columns on the screen. It is based on the CSC Full-Screen Display package, and can be easily customized for specific terminal/computer configurations. Data is entered into formatted screens under three file classifications, with computed values appearing in the fourth file. Any of the data entered can be independently updated. Utility programs allow any of the first three classifications to be used as part of another spreadsheet, and computed values can be used as initial values for another spreadsheet. The package is $100.

Computer Systems Consultants, 1454 Latta Lane, Conyers, GA 30207. Reader Service number 466.

TRS-80 Games

Acorn Software Products, Inc., 634 North Carolina Ave., S.E., Washington, DC 20003, is offering entertainment programs for the Model I and Model III. Included are Pinball, Duel-N-Droids, Everest Explorer, Invaders from Space, Gammon Challenger, Pigskin, Basketball, Space War!, Quad and Word Wars. Cassettes cost $14.95. Reader Service number 474.

All Atari Magazine

A special Atari issue of *Purser's Magazine* is available from Robert Purser, PO Box 466, El Dorado, CA 95623. The magazine features reviews of over 40 programs and an Atari software directory. It has 48 pages and costs $1 (newsprint) or $3 (betterstock paper). Reader Service number 476.

Word Processing For Apple III

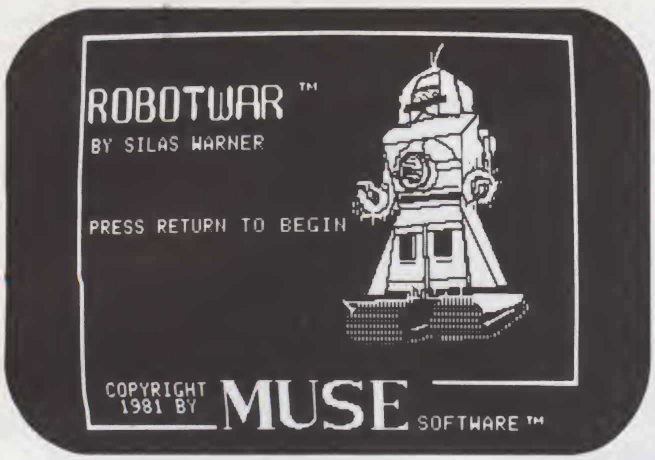
Write-On III is a new word processor from Rainbow Computing, Inc., 19517 Business Center Drive, Northridge, CA 91324. It has simple commands, such as C for change, I for insert, MB for move block and so on. Data can be inserted from the keyboard or from data files as a document is being printed. Up to 99 files can be merged to create large documents. The program can read, edit and print text files created by other programs. A formatted screen draft shows exactly how the printed text will appear. Price is $249. Reader Service number 469.

* * * * *

Another Apple III word processing program is available from Imagineering, Inc., c/o Adcast Advertising, 405 S. Farwell, Suite #10, Eau Claire, WI 54701. Most of Type-Righter's command functions require typing a single letter. Type-Righter features automatic justification as text is entered, global search and replace of words and phrases, moveable blocks of text, adjustable tabs and individual line centering. The display shows the user how the work will be printed. Pica or elite type can be selected. Type-Righter also includes automatic envelope addressing. Price is $195. Reader Service number 470.

Computer Literacy Gaming for the Apple II

RobotWar incorporates the logic of systematic computer programming into the fundamental strategy *of the game*—it's not a manual dexterity game. The RobotWar player writes a special Battle Language program which gives his or her robot its individual personality. This language controls such things as the robot's radar, laser cannon, speed and position. On the robot test bench, the player has a cybernetic window into the robot's mind to check that the program is performing as planned. Up to four robot opponents are chosen from those written by other players or



RobotWar, a battle of programming techniques, from Muse.

from the demonstration robots on disk. Completed robots can be stored in encrypted format on a friend's disk, so players can share robots without revealing their program secrets.

RobotWar requires an Apple II or Apple II+ computer *with 48K, Applesoft ROM* and disk drive running 3.2 or 3.3 DOS. Disk, documentation and RobotWar Club membership application are $39.95.

Muse Software, 330 North Charles St., Baltimore, MD 21201. Reader Service number 471.

Information Master Companion

Transit is a new utility that enhances the Information Master data management system. Transit will convert most Apple II data files into Information Master files, as long as the records of the data file contain the same number of fields. This lets you use data files from other software packages in new ways, and also lets you convert outdated software to more convenient format. The new Information Master file can be sorted, searched, calculated and printed in custom-designed reports. Files can be subdivided, appended and manipulated in many ways. Transit runs on the Apple II with 48K and disk drive. Price is $50.

High Technology Software Products, Inc., 8001 N. Classen, PO Box 14665, Oklahoma City, OK 73113. Reader Service number 477.

Strategy Games for Apple II

CE Software, 801 73rd St., Des Moines, IA 50312, has three new games for your Apple microcomputer. Mission Escape! combines action and strategy as you face storm troopers, robots and mindless drones, and attempt to escape from the enemy ship. Wall Street lets up to nine players buy and sell stocks in a hyperkinetic market. Courage, foresight and luck are needed to build your fortune, and bankruptcy may be only one bad decision away.

The Case of the Sultan's Pearl leads you through a castle, spotting clues and analyzing evidence to learn who killed the Sultan's guard and stole the sacred pearl. Act quickly or the sultanship will be lost. Each game costs $24.95. Reader Service number 480.

Overlay Linking Loader for Microcomputers

Lynx is an overlay linking loader for Microsoft's FORTRAN, COBOL and Macro-80. It features simple commands and a complete Help function, and will work with other language translators which produce Microsoft-compatible relocatable files. Lynx allows the construction of programs that use *all* available memory, including that used by Lynx itself, so programs that have reached the maximum size allowed by Microsoft's L80 linker can be increased a

minimum of 9K using Lynx without overlays. With overlays, many large programs currently running on minicomputers and mainframes can be modified to run on a microcomputer. Lynx reduces the memory required by dividing the program into overlay segments that are brought into memory from disk as needed. Parameter values can be passed to an overlay as though it were a subroutine or through common blocks. The system costs $250.

Westico, 25 Van Zant, Norwalk, CT 06855. Reader Service number 463.

Report Writing on CP/M

A combined numerical problem-solving and word processing program is available from Lifeboat Associates, 1651 Third Avenue, New York, NY 10028. T/Maker II produces charts and exhibits, and includes screen text editing controls for complete report writing. Reports can be prepared in letter format by merging a preprogrammed mailing list.

To create a table using T/Maker II, enter the data and define the relationships between rows and columns. The program will compute established equations and place answers in their appropriate positions. Any changes you make are automatically posted to corresponding rows and columns. The package costs $275. Reader Service number 467.

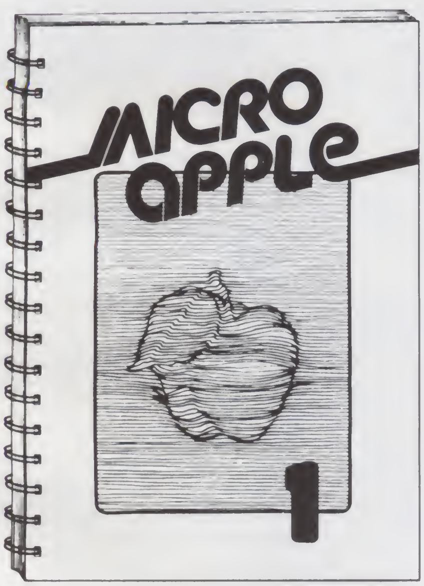
Payroll for Apple II

A new payroll program employs Apple Computer Company's "Run Time Module." This lets you run Pascal programs on the Apple II with DOS 3.3 without a special language card. The payroll handles 300 employees, 15 divisions and 30 deduction types. It computes all federal and state income taxes, plus other state and local taxes for all states. Tax formulas are built in, and updates are provided at low cost. The program prints checks, a check register, W-2 forms and all quarterly and summary reports. Two disk drives are required. Price is $395.

Broderbund Software, Box 3266, Eugene, OR 97405. Reader Service number 468.

Broderbund Software's Payroll package in Pascal.

GET MORE OUT OF YOUR APPLE WITH MICRO on the APPLE



MICRO/Apple

Over 30 Apple Programs on Diskette — For Less Than $1.00 Apiece! No Need to Type In Hundreds of Lines of Code!

224-page book and diskette $24.95*

* Add $2.00 for surface shipping. Massachusetts residents add 5% for sales tax.

MICRO's new book for Apple II users lets you

- Speed up programming in Applesoft and Integer BASIC!
- Add Apple II Plus editing features — at no cost!
- Round and format numbers accurately in business applications!
- Get lowercase letters and punctuation into Applesoft strings — at no cost!
- Do a shape table easily and correctly!
- Play the hit game "Spelunker"!
- And much, much more!

With MICRO/Apple 1, the first volume in our new series, you receive

- 30 choice articles from MICRO, The 6502 Journal (1977-80), complete with listings, all updated by the authors or MICRO staff,

plus

- 38 tested programs on diskette (13 sector, 3.2 DOS format, convertible to 3.3).

Ask for MICRO/Apple at your computer store or
Call Toll-free 800-227-1617, Ext. 564
In California, call 800-772-3545, Ext. 564
VISA and Mastercard Accepted

MICRO
P.O. Box 6502
Chelmsford, Massachusetts 01824

BOOK REVIEWS

Edited by Eric Maloney

Introduction to Pascal
Buyer's Guide to Business Systems
Micromatics
BASIC Scientific Subroutines
Small-Business Computers

Introduction to Pascal Including UCSD Pascal

Rodnay Zaks
Sybex, Inc.
Berkeley, CA
Softcover, $14.95

This book is probably the best 420 pages about Pascal on the market today. I haven't found another book that is as complete and well-written.

My enthusiasm stems not so much from the language described, but from the organization and choice of example programs. Zaks has avoided the temptation to write a textbook for advanced computer science students. The examples are clear and illustrate programming approaches you would use in writing business and scientific programs.

The author twice slips from the totally practical to the theoretical to illustrate some principles. The first is a program to encipher text with a simple substitution algorithm. At a time when experts cannot agree whether the National Bureau of Standards' complex data encryption standard is effective, I'm sure that a cipher program based on simple substitution of letters will not truly protect my privacy.

The second fall from grace in the book is the inclusion of about ten pages of text and program about binary trees with the following introduction: "Binary trees are used for many purposes. They are used in particular when encoding decision algorithms, or analyzing the structure of text or data."

If Zaks had told me instead that the examples illustrated an important use of recursion and dynamic data structures, I might have viewed the subsequent programs in a different light. The problem is that the program did nothing more than order a set of integers. It told me nothing about decision algorithms or analyzing the structure of text or data.

However, to be fair, I must admit that my concentration on programs for the business environment has prejudiced me against tree structures (or any other data structure) which reside only in memory and are vulnerable to accidental resets.

With the exception of the two cases I have noted, the rest of the book has examples which are well suited to the text. Both the text and examples concentrate on problems you are likely to face in writing programs for real-world applications.

The material on UCSD Pascal is handled well, usually with a discussion and examples at the end of the chapter. I like having the UCSD material separated from the rest of the text, as it cuts down my search time when I'm looking for a special feature of that dialect of Pascal.

The size of the book and the complexity of some of the later examples may intimidate the Pascal novice. However, a short section at the front of the book tells the novice how to best use the book. Also, a number of exercises (with solutions at the back of the book) let you test your comprehension.

The more advanced Pascal user will find the book an excellent reference source. The table of contents and list of illustrations span eight pages and will allow you to find an example which will help you understand almost any programming problem you may encounter. There are also 11 appendices covering the details of standard and UCSD Pascal.

In summary, I think this is one of the best books available on Pascal. The author has kept his feet firmly planted in the arena of real-world applications. The prose is, though rather formal, clear and concise. The examples are interesting and, with only a few exceptions, illustrate problems which must be solved by programmers working for more than university grade points.

Mark J. Borgerson
Corvallis, OR

Business System Buyer's Guide

Adam Osborne, with Steven Cook
Osborne/McGraw-Hill
Paperback, $7.95

Where do you go if you're a business owner who realizes his business can make more profit if it uses a microcomputer, but who also understands that he knows next to zilch about buying one of the darn things and much less about operating one?

You go out and buy—and then read—

this book.

The first line in the *Business System Buyer's Guide* says it all: "This book is written for nontechnical people who are interested in small computer systems."

The basic tone of the book makes you feel like you're sitting across a desk from Osborne and Cook. You've told them you're in business doing such-and-such and want a computer to handle, you think, this and that, and you kind of figure you can spend so much money. So what should you buy?

The best part is that they do *not* tell you, "Go out and buy an Apple II," or "Buy the new Osborne computer," or "If you can get monthly payments buy at Radio Shack."

Instead, *they make you think*.

They ask you to sit down and define your business procedures and problems, to put on paper exactly what you do now and how it's handled. You can't determine what programs you need until you define your own methods of doing things. They take it a step further, too, by helping you flowchart your own business/accounting procedures, and you can see—perhaps for the first time—a graphic picture of how your business works.

As Osborne and Cook note, "Your goal is to ensure that all functions (in your business) have been accounted for." And you must take these steps—writing down what you do—before you even *think* about buying your computer.

They try to simplify things as they go. For instance, it's hard to evaluate a complex accounting system, with such things as accounts receivable and accounts payable and cash disbursements and payroll and your general ledger all connected, all flowing with information from one area to all the others. You can perhaps see what's being done now in your own set of books for all these things, but all the entries are made by hand, and with the computer they might be done automatically. How in the world could you ever trace a transaction all the way through?

Osborne and Cook tell you that you can get as complex as you wish, but "when using small microcomputer systems and simple data processing programs, you may find that reentering the same information twice is simpler and quicker than feeding information from one function to the next."

What they don't mention is that this method of entering data also gives the user (your bookkeeper, or yourself) a form of real feedback—you can see that you've posted something to your accounts receivable file, you can see that you've posted a check both to accounts payable and to cash disbursements. You know (because you've seen) that the data has been entered and stored properly.

The authors work to simplify other matters, too. They suggest that you think of memory and disk storage not in terms of bytes, but in terms of characters. Your name is John, and that's four characters, and so on. To "think of stored information in terms of files that you might keep in a filing cabinet." To "think of the display as . . . an indexing system which can instantly pull a ledger card from a filing cabinet and place it on your desk." This is people talk, not computer talk.

They suggest that, as a final step in examining your business, you write down what output you'll require and the format you'd like to have it in.

All this before even thinking about which computer you want to buy.

They go on to give a description of programming languages, and a lot of good, easy-to-understand data on what to look for in your hardware—keyboard configurations, monitors, memory, storage media, etc.

Now, since the average reader of this book will not be doing his own programming, Osborne and Cook suggest that you start by evaluating the available software in terms of what you've learned about your own business. You now know (from the examples you've run through) exactly how your business works on paper—so the starting point is not to pick out a brand of microcomputer and buy one, but to determine which computer you will get based on what software is available now that will handle the procedures you need.

> If you're a nontechnical person who's interested in a small computer system, you won't find a better investment than the Business System Buyer's Guide.

To help the reader discover what is available, a number of charts show a complete listing of available software packages, which systems they work on, where to buy them and their cost.

I noticed that a few brands of equipment were missing from the descriptions—notably, Cado, IBM and Hewlett-Packard. So I called Adam Osborne and asked about it. He was trying, he said, to keep the basic systems under $10,000. This seems a realistic figure, perhaps about what the small businessman (for whom this book was written) could and should spend.

The only real deficiency is in the list of available printing equipment. For example, only one Centronics model is shown, out of more than perhaps a half-dozen you could hook to your microcomputer.

This book would have been an easy opportunity to push Osborne's new microcomputer, which is brand new and looks terrific on paper. But, to their credit, the authors gave this new computer less space than any of the others.

If you're a nontechnical person who's interested in a small computer system, you won't find a better investment than the *Business System Buyer's Guide*. Even if it doesn't fit exactly, and you already talk computer, or have your own system, or can do some programming, you'll learn a lot from this terrific book.

Gregory Glau
Prescott, AZ

Micromatics

Steven K. Roberts
Scelbi Publications
Hardback, 190 pp.

This book was written for the electronics technician who doesn't need to know (or possibly care) how the various internal timing and control signals relate. It was written for the electronics technician who must use a microprocessor in an integrated system to interface to the outside world. In other words, it picks up where the others leave off—at the outside world.

While the microprocessor is not ignored, it is de-emphasized, and viewed as one component in a system, rather than as an all-important device to which the system is appended.

The author takes an excellent pedagogical approach to the subject matter by relating the reader's experience, or "internal model of the world," to the text. While he assumes that the reader is familiar with basic electronic terminology, he also assumes that the reader knows little or nothing about microprocessors.

The author introduces in chapter 2 a computer-controlled stereo cassette deck. By first describing the basic required functions in block diagram form, the author effectively details the system requirements without frustrating the noncomputerist reader with a lot of computer jargon.

Because it would be naive to expect any book on this subject to make an adequate presentation and not establish a basis for the reader's comprehension of how a computer system operates, chapter 3 gives the reader a first look at the hardware. Buses are explained, along with logic states, memory types and organization, control signals and the like. At the end of the chapter, the previous information is related back to the real-world example of the cassette controller.

Chapter 4 looks at the internal design

This book (Micromatics). . . .
is a welcome change
from most digital books
full of jargon and schematics

of the 8080. The 8080 was chosen, as opposed to more advanced chips, because it represents one of the major families of microprocessors, and most of the others are closely related. However, in subsequent chapters the Z-80 is used as a design example. While the transition from 8080 to Z-80 is relatively easy for the experienced reader, the author should have used one chip or the other.

While Roberts does not go into a great deal of detail on the internal design, the chapter provides a good foundation for the software discussion, which begins in chapter 5. The author alludes to machine-language instructions using binary, hexadecimal and decimal numbering systems, and then quickly moves to assembler language programs as the basis for the control network programming to be done throughout the text. Through judicious use of flowcharts and lots of written description, even the uninitiated reader can understand the examples. Only a few representative instructions are explained and used in the chapter, so the reader is not overwhelmed with details of instructions, mnemonics and the like.

Chapter 6 includes two actual programs, along with detailed explanations of what is being done by the programs and why. The real world is brought back into the picture with a program for the cassette controller. Illustrations of the controller hardware greatly aid comprehension. Since the cassette controller is basically a data collector, and since most processor-based industrial control devices are basically data collectors, an industrial data collection network is discussed in chapters 7 through 9.

The concept is discussed in chapter 7 with the aid of many sketches and block diagrams. The background (to which a majority of readers can probably relate) is developed into hardware, complete with a very detailed schematic diagram. While the schematic is impressive, I'm not quite sure why it was so detailed (including pin numbers, discrete component values, etc.). It seems very unlikely that a reader would actually undertake construction of the controller.

Having presented the hardware in chapter 8, the author presents a discussion of the software in chapter 9 (the entire assembly-language program, complete with copious comments, is presented in Appendix C). The book's continuity stumbles a little here. All mnemonics used in previous assembly-language discussions have been based on the 8080 instruction set. Suddenly the Z-80 instruction set is introduced, along with a conversion table (from 8080 to Z-80 mnemonics) in Appendix A. There would have been less chance for reader confusion if the Z-80 had been used as the basis for all discussion.

That complaint notwithstanding, the author has done well to put the program listing out of the way in an appendix, so that the logical flow of the chapter is not interrupted. The chapter presents an excellent discussion of system software operation. The discussion is specific enough for the interested reader to be able to understand actual operation of the various program modules, but not so specific that the reader loses sight of overall system operations.

The concept of real-time operation, developed in the preceding three chapters, is elaborated on again with a different example in chapter 10. Some interesting concepts for industrial control are presented which the interested reader will probably want to pursue further.

Having discussed only industrial applications thus far in the book, the author presents his personal system in chapter 11. At first, a discussion of personal computers seems out of place, but given the intended audience, it helps to give the reader an overview of how the same microprocessor can be used outside the factory.

As any owner of a troublesome microprocessor system can attest, service seems to be an afterthought in any design. Chapters 12 through 17 are devoted exclusively to service—and how to design for it. These chapters alone make the book required reading for anyone about to design a digital system.

This book will probably have minimum appeal to the average hobbyist, but will appeal to anyone with a background in controls who needs to interface to (or design) a digital system. It is a welcome change from most digital books full of jargon and schematics.

J. C. Hassall
Blacksburg, VA

Basic Scientific Subroutines, Vol. 1

F. R. Ruckdeschel
Byte Books, McGraw-Hill
Hardcover, 316 pp., $19.95

One of the major untapped markets in the microcomputer industry is the scientific user. Most research laboratories, even small ones, will have one large mainframe computer and several smaller ones, usually for data acquisition and analysis. Yet there is a distinct and serious need for applicable hardware and software. Most scientists and technicians spend much of their time creating software.

In an attempt to help solve this serious problem, Dr. Ruckdeschel has written this first in a series of books covering some of the more commonly-used mathematical techniques in the sciences. Volume 1 covers topics such as printer plotting routines; arithmetic involving complex variables, vector manipulation, matrix algebra, operations and applications; various random number generators; and series approximation of transcendental functions.

The book is not written for the casual computer user, the games groupie or even the businessman with a computer. It is specifically written for the engineer, scientist, computer scientist or science student. It contains a series of subroutines in the BASIC language, with demonstration calling programs and sample runs, and a complete description of the mathematical principles and types of applications (with examples) of each program.

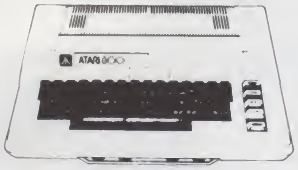
To make full use of the book, a prior knowledge of the mathematics discussed is essential. The user should be familiar with complex algebra, vectors and matrices. A knowledge of calculus, while not necessary, would be helpful.

Most of the subroutines in the book are those that a scientist or engineer usually has to re-invent for himself over the years. As a physicist, I have had to develop the printer plot routine, determinant of a square matrix, solution of simultaneous equations routine, and various Taylor series approximations. Until now, my primary standard reference on the subject has been a worn and dog-eared copy of Bevington's *Data Reduction and Error Analysis for the Physical Sciences*, which has all of its computer programs written in the old FORTRAN II, which hardly anyone uses anymore. Since most micros use BASIC, the present volume will be much more valuable.

Programs are presented as a package with consecutive line numbers, to be appended onto an analysis program as needed. Many of the subroutines call other subroutines within the package, so the whole package is needed. The user program then calls up subroutines in the package. The BASIC used in this book is Microsoft BASIC. Copies of the subroutine package are available on disk or tape from the author.

Many programs that would be valuable, such as a general regression-analysis, chi-square test, correlation analysis and numerical integration, are not presented. I can only hope that they will be given

we beat the price...



800™ $779

WITH ONE YEAR EXTENDED WARRANTY

ATARI®
Computers for people.™

815 DUAL DISK DRIVE

$1199

ATARI SOFTWARE

CX401 General Accounting	$399.00
CX402 Accounts Receivable	$399.00
CX403 Inventory Control	$399.00
CX404 Word Processor	$119.00
CX405 PILOT	$68.00
CX413 MICROSOFT BASIC	$68.00
CX4101 Invitation to Programming 1	$17.00
CX4102 Kingdom	$13.00
CX4103 Statistics	$17.00
CX4104 Mailing List	$17.00
CX4105 Blackjack	$13.00
CX4106 Invitation to Programming 2	$20.00
CX4107 Biorhythm	$13.00
CX4108 Hangman	$13.00
CX4109 Graph It	$17.00
CX4110 Touch Typing	$20.00
CX4111 Space Invaders	$17.00
CX4112 States & Capitals	$13.00
CX4114 European Countries & Capitals	$13.00
CX4115 Mortgage & Loan Analysis	$13.00
CX4116 Personal Fitness Prog.	$59.00
CX4117 Invitation to Programming 3	$20.00
CX4118 Conversational French	$45.00
CX4119 Conversational German	$45.00
CX4120 Conversational Spanish	$45.00
CX4121 Energy Czar	$13.00
CX4125 Conversational Italian	$45.00
CX6001 U.S. History	$23.00
CX6002 U.S. Government	$23.00
CX6003 Supervisory Skills	$23.00
CX6004 World History	$23.00
CX6005 Basic Sociology	$23.00
CX6606 Counseling Proced.	$23.00
CX6607 Principal of Act.	$23.00
CX6608 Physics	$23.00
CX6609 Great Classics	$23.00
CX6010 Business Comm.	$23.00
CX6011 Basic Psychology	$23.00
CX6012 Effective Writing	$23.00
CX6014 Principals of Econ.	$23.00
CX6015 Spelling	$23.00
CX6016 Basic Electricity	$23.00
CX6017 Basic Algebra	$23.00
CX8106 Bond Analysis	$20.00
CX8107 Stock Analysis	$20.00
CX8108 Stock Charting	$20.00
CXL4001 Education System Master	$21.00
CXL4002 Basic Computing Language	$46.00
CXL4003 Assembler Editor	$46.00
CXL4004 Basketball	$24.00
CXL4005 Video Easel	$24.00
CXL4006 Super Breakout	$30.00
CXL4007 Music Composer	$45.00
CXL4009 Chess	$30.00
CXL4010 3-D Tic-Tac-Tow	$24.00
CXL4011 Star Raiders	**$32.00**
CXL4012 Missle Command	**$32.00**
CXL4013 Asteroids	**$32.00**
CXL4015 TeleLink	$20.00
Compuhome	$74.95
Visicalc	$149.00
Letter Perfect (Word Processor)	$119.00
Source	$89.00

Atari® Peripherals:

400 16K	$329.00
410 Recorder	$59.00
810 Disk Drive	$449.00
815 Disk Drive	$1199.00
822 Printer	$359.00
825 Printer	$629.00
830 Modem	$159.00
850 Interface	$139.00

Atari® Accessories

New DOS 2 System	$21.00
CX70 Light Pen	$64.00
CX30 Paddle	$18.00
CX40 Joy Stick	$18.00
CX853 16K RAM	$89.00
Microtek 16K RAM	$75.00
Microtek 32K RAM	$169.00

CBM 8032 $1099

4016	$799.00
4032	$999.99
8096	$1795.00
CBM4022 Printer	$629.00
Tally 8024	$1699.00
CBM C2N Cassette Drive	$69.00
CBM4040 Dual Disk Drive	$999.00
CBM8050 Dual Disk Drive	$1349.00

commodore

VIC 20

$259

Vic-TV Modual	$19.00
Vic Cassette	$69.00
Vic Disk Drive	$ Call
Vic 6 Pack program	$44.00

CBM Software

WordPro3 Plus	$199.00
WordPro4 Plus	$299.00
Commodore Tax Package	$399.00
Visicalc	$149.00
EBS Accts. Rec./Inventory Interactive Syst.	$595.00
BPI General Ledger	$329.00
OZZ Information System	$329.00
Dow Jones Portfolio	$129.00
Pascal	$239.00
Legal Time Accounting	$449.00
World Craft 80	$289.00
Word Check	$180.00
Create-A-Base	$180.00
Power	$89.00
Socket-2-Me	$20.00
Jinsam	Call

Disks

CX8100 Blank Disk (5)	$22.00
Sycom Blank Disk (10)	$29.00
Maxell Blank Disk (10)	$36.00
Maxell Blank Disk (10)	$46.00

Printers

Epson MX-70	Call for Prices
Epson MX-80	Call for Prices
Epson MX-80 FT	Call for Prices
Diablo 630	Call for Prices
Nec 5530	$2495.00
TEC 1500 Starwriter 25cps	$1495.00
TEC 1500 Starwriter 45cps	$1995.00

No Risk, No Deposit On Phone Orders, COD or Credit Card, Shipped Same Day You Call *

* on all in stock units

IN PA. CALL: 717 - 323-7921

(800) 233-8950

To Order:

Phone orders invited (800 number is for order desk only). Or send check or money order and receive free shipping. Pennsylvania residents add 6% sales tax. Add 3% for Visa or M.C. Equipment is subject to price change and availability without notice. Please call between 11 AM & 6 PM.

COMPUTER MAIL ORDER ✓384
501 E. 3RD ST., WILLIAMSPORT, PA 17701

✓See List of Advertisers on page 210

space in future volumes of the series.

Readers may find some of the programs less than fully useful. Series approximations are given for such functions as sine, cosine, log and exponential. Most extended BASICs will have these functions as part of the interpreter or compiler. They will, of course, be useful when they are not otherwise present. There is a discussion of execution times of these series versus the machine coded form in the interpreter, and it seems that the machine form is always faster. However, there is also a discussion of precision in the shortened form of the series, where the series is truncated to two or three terms, and the coefficients optimized for best fit over some small range. In this case, the series approximations compare very well, and may actually be faster than the machine form, especially in a compiler environment. The increase in speed is at a sacrifice in accuracy, typically to about 0.1 percent, as opposed to ten-digit accuracy in machine forms.

The only real problem with the book is the price. At $19.95, many scientists will probably be willing to spend a little extra time writing their own routines. If the series runs to three volumes, as I expect that it will, many scientists will find almost $60 to be a substantial investment. I would suggest to McGraw-Hill and Byte Books that if they were to market the book in softcover at $8.95, their sales would probably triple.

Dr. Gordon W. Wolfe
University of Mississippi
University, MS

Small Computers for the Small Businessman

Nicholas Rosa and Sharon Rosa
dilithium Press
Paperback, 340 pp., $12.95

The Rosas' book is a well-written, comprehensive, all-in-one manual for the businessman interested in both computing and succeeding. It will help get the absolutely new-to-computers businessman off to a good start in what computers can do for him, and are currently doing for his competitors.

After whetting his appetite, the businessman who is new to computers will probably want to browse through the glossary. Terms like dual-in-line, dynamic RAM, turn-key system and loop are explained in clear, well-written English. Combining the trip through the glossary with a desire to figure out how a computer will help his particular business will provide the reader with an idea of which chapter to go into next.

For instance, if the reader's business needs word processing, there is an excellent chapter which first explains what word processing is and what it can do to help alleviate business problems. But the chapter doesn't stop there—it goes a step further and clearly explains how word processors function, what kind of word processing equipment a small business might need, and most importantly, how to go about learning how to use the tool once it is purchased.

Similar chapters are available for businessmen with preconceived notions of needing timesharing or a minicomputer or in-house software because the competition uses them. How to intelligently go about shopping for and making purchase decisons about these items and services also is discussed.

More advanced or knowledgeable microcomputer users, such as a hobbyist who uses a black box computer he did not personally build, will also be interested in what the Rosas' book has to offer. Most microcomputer owners are intimately familiar only with their own system. The Rosas contrast, and point out advantages to, many varied systems. They also go into sufficient technical detail to give computer users who seldom, if ever, peek inside at the hardware a chance to understand both how the hardware functions and what makes the difference between good and bad hardware for various business purposes.

On the whole, this book is one of the clearest, most easily digested overviews for the business-oriented person looking for some explanations into the mysteries of small computers. As has so often been pointed out, the businessman who decides to "wait a while until the microcomputer market straightens itself out" may be making a decision to go out of business if his competitor reaches an understanding first.

Kathleen Shoemaker
Blacksburg, VA

New and Noted

The Alternate Source: Volume I. Alternate Source magazine, 1806 Ada St., Lansing, MI 48910. Softbound, 300 pp., $14.95 plus $2 postage and handling. First six issues of magazine devoted to advanced applications and programming for the TRS-80.

The Analytical Engine, revised edition. By Jeremy Bernstein. William Morrow & Company, Inc. Softbound, 132 pp., $4.95. This introduction to computers is an update of the 1964 edition.

Beyond Games: System Software for Your 6502 Personal Computer. By Ken Shier. Byte Books/McGraw-Hill. Softbound, 433 pp., $14.95. "A comprehensive and definitive guidebook for owners of 6502-based personal computers. *Beyond Games*. . .offers a self-contained course in structured programming and top-down design."

Digital Counter Handbook. By Louis E. Frenzel, Jr. Howard W. Sams & Co., Inc. Softbound, 264 pp., $10.95. The basics of how digital counters work.

50 Programs in BASIC for the Home, School & Office. By Jim Cole. ARCsoft Publishers. Softbound, 80 pp., $9.95. Ready-to-run programs in Pocket-BASIC "for businessmen, teachers, students and hobbyists."

How to Design and Build Your Own Custom Robot. By David L. Heiserman. Tab Books. Softbound, 462 pp., $12.95. How to plan, put together and program a custom-designed artificial intelligence machine.

How to Get Started with CP/M. By Carl Townsend. dilithium Press. Softbound, 128 pp., $9.95. The title says it all, doesn't it?

Interfacing Microcomputers to the Real World. By Murray Sargent III and Richard L. Shoemaker. Addison-Wesley Publishing Co. Softbound, 288 pp. Emphasis is on the Z-80.

Introduction to 8080/8085 Assembly Language Programming. By Judi N. Fernandez and Ruth Ashley. John Wiley & Sons, Inc. Softbound, 306 pp., $8.95.

Microsoft FORTRAN. By Paul M. Chirlian. dilithium Press. Softbound, 334 pp., $14.95. Introductory text intended for first-time users.

Pascal Programming for the Apple. By T. G. Lewis. Reston Publishing Company, Inc. Softbound, 234 pp., $12.95. Discusses features unique to UCSD Pascal.

Programmer's Guide to the 1802. By Tom Swan. Hayden Book Company, Inc. Softbound, 158 pp., $7.95. Assembly language primer with assembler. Intended for the beginner.

Robot Intelligence with Experiments. By David L. Heiserman. Tab Books. Softbound, 322 pp., $9.95. Experience artificial intelligence "by watching robot thinking at work on your own home computer."

67 Ready-to-Run Programs in BASIC: Graphics, Home and Business, Education, Games. By Wm. Scott Watson. Tab Books. Softbound, 182 pp., $6.95. *Nobody* comes up with longer titles than Tab. Programs use TRS-80 Level I BASIC.

6809 Microcomputer Programming and Interfacing with Experiments. By Andrew C. Staugaard, Jr. Howard W. Sams & Co., Inc. Softbound, 272 pp., $13.95.

Taming Your Computer. By Jerome Kanter. Prentice-Hall. Hardbound, 246 pp., $17.95. How business and professional people can get the most from their computer systems.

✓See List of Advertisers on page 210

SS-50 Interface

Inexpensive Serial Board From CSC

CSC SI-1 Serial Board

Every SWTP user who has acquired a high speed printer, terminal, or modem knows one of the design limitations of the MP-S serial interface: it does not provide the necessary handshaking signals for high speed serial transmission. Some modifications have been described in *Microcomputing* ("Dynamic Duo" by Dr. P. Vijlbrief, May 1980, p. 188), but these require additions to the MP-S. Gimix offers a very good single serial port, but the $88 price will put it out of the range of many home users.

Computer Systems Consultants (1454 Latta Lane, Conyers, GA 30207) provides a painless alternative. Their SI-1 serial board provides two RS-232 signals for transmit-and-receive, optional TTL signals and the standard handshaking signals RTS, CTS and DCD. The SI-1 is designed around the 6850 ACIA and occupies a single 30-pin I/O slot on the SWTP.

The CSC bareboard is priced at $10. While a kit is not available, careful shopping for parts will keep the total cost at around $25. The board is single-sided with no solder masking or labels on the component side. There is no component placement layout other than that provided on the foil traces. In several cases parts must be traced from the schematic to ensure proper placement. Several diodes are required for RS-232 operation, but none is specified. These can be any of the general-purpose small signal types such as the 1N4148.

The 1488 IC can be damaged if your system's – 12 V line runs high. You must insert a 12 V zener diode and resistor between the – 12 V bus and the 1488 IC. There is ample prototyping area on the board so this will create no problems. Otherwise the documentation is clear. It provides assembly instructions and a schematic, and lists the available options of the board.

The SI-1's real advantage is its versatility. Options can be selected by jumpers. In the future, if your needs change, it is a simple matter to change a few jumpers and reconfigure the board. You can select either a single set of TTL channels or dual RS-232 channels. Since there is a single ACIA on one port, think of it as a dual-port interface as used on the new 6809 systems. The secondary data in and data out lines provide the same information as the primary channel. This lets you use two terminals, keyboards or a terminal and a printer on the same port with the same data.

Because of the flexibility, the SI-1 has many jumpers. Plan ahead when assembling the board and place the jumpers in an orderly manner.

All I/O connections go to a single 16-pin DIP header. This makes it easy to attach a D connector on the back of the computer and does not require an expensive double D connector extension.□

Dennis Doonan
Racine, WI

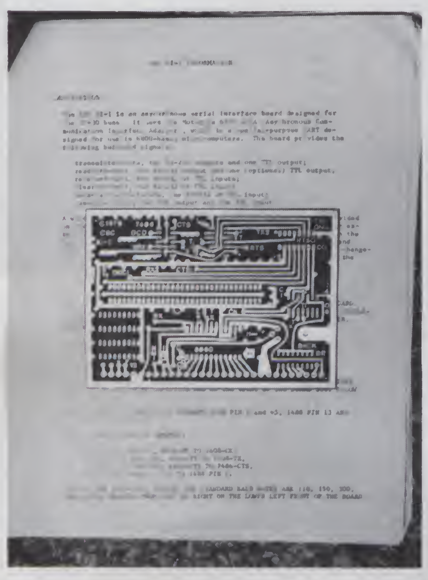


Photo 1. The CSC SI-1 bareboard with the documentation. This is what is received for $10.

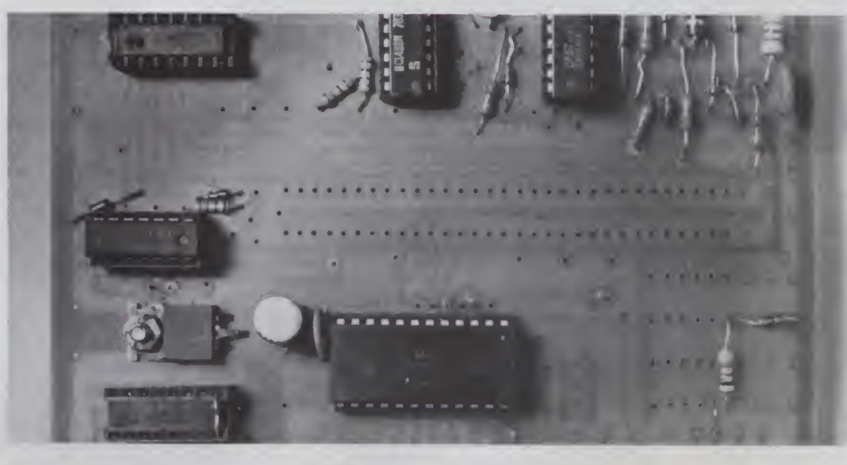


Photo 2. The completed CSC SI-1 interface set for RS-232 operation. The jumpers have been placed on the back side. The empty socket in the upper left corner is the 16-pin DIP header for the I/O connections.

KB BOOK NOOK

FOR TRS-80* OWNERS ONLY!

THE NEWEST PERIPHERAL FOR THE TRS-80*

The **Encyclopedia for the TRS-80***, published by Wayne Green Books, is a ten-volume reference work that is absolutely packed with programs and articles carefully selected to help you make the most of your microcomputer. The volumes of the Encyclopedia can be considered an extension of the documentation that came with the TRS-80*, offering a continual supply of information and ideas that will help you realize the full potential of your TRS-80*. Each volume contains material on programming techniques, business, games, tutorials, utilities, and interfacing.

Unlike conventional encyclopedias, the **Encyclopedia for the TRS-80*** will never become stale or out of date. That's because the volumes of the Encyclopedia are being issued one at a time, over a period of months. This means that each new volume will reflect the latest developments and discoveries, making this a living encyclopedia for TRS-80* users.

Here's a sample of what's included in Volume 1:

- Four Graphics Methods. Improved techniques for dealing with the TRS-80's low resolution graphics.
- TTY Interface. Build an interface, hookup a Teletype to a TRS-80.
- The President Decides. A thrilling simulation that lets you make decisions as if you were President of the United States.
- Measure Instructional Effectiveness. Many programs teach or tutor: this one helps determine the effectiveness of the lessons.
- The Invoice. A program for small businesses which produces custom invoices.
- Punch Out Your Disks. Double your disk storage in seven easy steps.

*TRS-80 is a trademark of the Radio Shack Division of Tandy Corp.

ORDERING INFORMATION

HARDCOVER SERIES			SOFTCOVER SERIES		
EN8101	VOL. 1	19.95	EN8081	VOL. 1	10.95
EN8102	VOL. 2	19.95	EN8082	VOL. 2	10.95

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to Kilobaud *Microcomputing* Book Department • Peterborough NH 03458. Be sure to include check or detailed credit card information.

No C.O.D. orders accepted. All orders add $1.00 handling. Please allow 4-6 weeks for delivery. Questions regarding your order? Please write to Customer Service at the above address.

PRICES SUBJECT TO CHANGE WITHOUT NOTICE

FOR TOLL FREE ORDERING CALL 1-800-258-5473

Complete your computer installation with our 24 Hour Analog Computer Clock

This quartz accuracy, bold face clock (15" in diameter) is GUARANTEED free of defects for 18 months. It's battery operated... and we include the battery for only **$38.95**

Call toll free: **(800) 645-3955**
Charge it on Mastercard or Visa
In NY State, call (516) 378-5373

or write to: VENTURA CORPORATION ✓114
1597 Grand Avenue, Baldwin, NY 11510

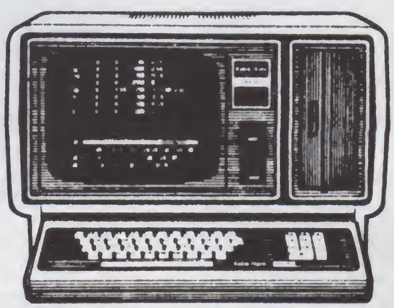
Also check our low prices on Qume, Diablo & Verbatim diskettes, library cases, computer ribbons, print wheels and many other quality accessories for the hobbyist or businessman.

SAVE $$$ SAVE $$$

TRS-80®

MICROCOMPUTERS

CALL US...
SAVE MONEY



Model II
64K
$3325.00
Full 3 Month Warranty

CALL COLLECT:
512 - 689-5536
Master Electronics, Inc. ✓72
154 N. 5th, Raymondville, Tx. 78580

master charge

Form F48 Provided
Standard Warranty On Merchandise

VISA

Authorized TRS-80 Dealer, Store #F-723

PERSPECTIVES

(from page 242)

cause managers to obtain larger equipment than is needed for the job at hand and to spend unnecessarily. Wasting money somehow never hurts prestige; it actually enhances it.

That works against small computers. They are not as pretentious as the larger systems; while at the same time they are not compact enough for desktop use. (You cannot clutter up a desk with a microcomputer, monitor, disk drive and printer, and leave room for much else.)

Those who obtain the larger equipment may be discarding it sooner than they expect, and the knowledgeable manager will pick up his larger "prestige" system cheaply later on, while he obtains his microcomputer for that heavy work in the back office now. As the costly equipment becomes quickly outmoded, prestige value will soon disappear and the costly equipment will soon become an embarrassment.

It is probably inevitable that you are going to see microcomputers in front offices before you see them elsewhere. Managers will slowly learn that the microcomputer is more of a practical production machine than it might seem. The heavier the clerical load, the more work it will do. The paybacks will be greatest in the back offices and in the production areas, which are most neglected by central data processing systems. Whereas out-front microcomputers may be a convenience and a conversation piece, elsewhere they will become part of the machinery of productivity. They are already being used to run machines, monitor lighting and heat, and handle other chores, in addition to desk work.

At a time when productivity improvement is beginning to become a national priority, those companies which make good use of micros, both in front and back offices, will get their share of respect without wasteful expenditures.

Some of the obvious benefits of microcomputers are hardly evident to a casual observer. The most mundane applications may save the most money. For example, used with a printer, what other office machine can handle all the work of a spreadsheet or ledger sheet and produce the final copy in typed format? It is surprising that microcomputers are not being used much for such applications.

All of this is quickly recognized by the people who do the actual work, but it may escape the people who obtain the equipment, even if they count the keystrokes and prove the savings to themselves.

The bottom line here is that micros are labor savers like ordinary adding machines were. But labor savings are not always popular; prestige is.

Since microcomputers will do away with drudgery jobs, managers who see their prestige reflected in the number of people who work for them will quickly conclude that the micros have no application to their particular operations. Likewise, some manufacturers who hold allegiance to larger, older data processing equipment are deeply committed to not being displaced by the microcomputer.

In a few years microcomputers will be as commonplace as the telephone.

The Choice Is Yours

Many data processing people are really captives of the larger companies which supply or rent them equipment, and, above all, who supply them service. Often the equipment is not compatible with anyone else's system. (Lack of compatibility of equipment is another situation which can discourage a manager from going to a new microcomputer.)

When larger systems are eventually displaced, jobs will change. The old programming languages will be replaced by languages which use the programmer and the computer more efficiently. Good programmers are already busy learning the new languages.

Another obstacle to wider use of microcomputers is the feeling among managers that data processing systems never saved any money. This might go back to programming done in too much of a hurry to meet a deadline, by someone under more pressure to get the job done than to reduce work for others, and who was probably not trained as an efficiency or industrial engineer. These conclusions, however, also ignore the effect of the superior information which comes from data processing upon the success of the company.

It is true that large data processing systems require costly staffs and facilities. It is also true that managers may not get the full value out of any computer system. A manager needs to buy and use his system wisely. One benefit of microcomputers is that they do not require expensive facilities and staffs.

Advances in equipment may catch a buyer off guard. Plenty of people have bought equipment from companies which no longer exist. Buyers of any computer system need to be especially careful. The safest haven is the microcomputer, which represents the least investment.

It is easy to think you need a lot more equipment than is actually necessary to do the job, based on out-of-date (last year's, or even last month's) information.

A purchaser of equipment who has done his homework, and who is qualified to make a good choice, may wait six months to a year, while a manager who suddenly catches a glimpse of one particular application may be swept off his feet and spend far more than is necessary to do far less than he should be doing.

Another manager may not obtain any equipment at all. There are many reasons to avoid a decision. A lot of people are going to make bad choices, and at a certain level a manager may not be popular after a bad choice. The choice is undeniably difficult; I struggled with the decision six months before I bought a computer just for my own needs.

During this shakeout period for microcomputers, some manufacturers will not be here next year, when new companies will be introducing new equipment. Yet the manager who fails to act will do the most damage to his company.

When I found I needed to buy a computer to stay literate and competent as an industrial engineer (this field is rapidly changing), I resigned myself to spending $1200–$1400 for the microcomputer, plus more later for a printer. I began to modify our household budget in anticipation. Finally it became possible to obtain both small computer and printer for under $1000. With careful programming it will serve well, but I realize this equipment may be obsolete in five years.

For those who are about to purchase a microcomputer, whether for themselves or for their businesses, the safest purchase will be made after a good deal of reading, a course in BASIC, some work with somebody else's equipment, attending a few shows and plenty of conversation with others.

You should discount any game image you may have about the company that manufactures a microcomputer and go by their reputation with the people who know about their equipment, especially teachers, who are in general better informed than people in industry, because they use them in the classroom.

In a few years microcomputers will be as commonplace as the telephone and the choices will not be so hard. Most people cannot afford to wait that long if they do not want to become obsolete in their line of work. In a few years another generation will be out of school to replace anyone who has not kept up to date.

Managers with imagination, vision and perseverence will not be held back. One such manager solved his problem neatly by sending most of the people in his department, including himself, to a course in how to use microcomputers. Then, to get things moving, he leased two computers. He will not become obsolete, I suspect.□

Best Prices and Delivery!

SUPERBRAIN
by Intertec

Self-contained computer with dual disks and two RS232C ports. Complete with CP/M 2.2.
64K Double Density, List $3495 . . . . $2869
64K Quad Density List $3995 . . **SPECIAL $3395**

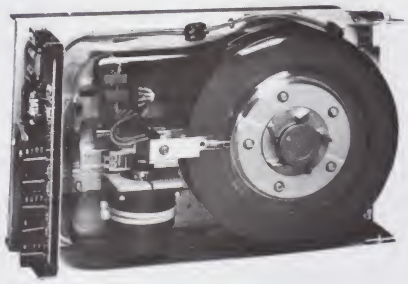
VIDEO TERMINALS
INTERTEC EMULATOR, List $895 . . . . . . $ 749
INTERTUBE III, List $895 . . . . ONLY 749
SOROC IQ 120, List $995 . . . . SPECIAL 729
SOROC IQ 130, List $699 599
SOROC IQ 135, List $1095 749
SOROC IQ 135 G, List $1195 799
SOROC IQ 140, List $1495 1149
HAZELTINE ESPIRIT, List $695 . . . . . NEW 595
HAZELTINE 1410, List $900 . . . . CALL
1420 . . . . CALL
1500, List $1225 . . . . CALL
1510, List $1395 . . . . CALL
1520, List $1650 . . . . CALL
TELEVIDEO 910, List $695 . . . . NEW CALL
912C, List 950 . . . . CALL
920C, List $1030 . . . . CALL
950C, List $1195 . . . . NEW CALL

PRINTERS
ANADEX DP-8000 . . . . $ 849
DP-9500, List $1650 1349
DP-9501, List $1650 1349
PAPER TIGER IDS-445, List $995 695
PAPER TIGER IDS-445G,
w/graphics op., incl. buffer, $1195 789
PAPER TIGER IDS-460, List $1094 919
PAPER TIGER IDS-460G, List $1394 1139
PAPER TIGER IDS-560, List $1295 . . . . . $1089
PAPER TIGER IDS-560G, List $1395 . . . 1169
NEC Spinwriters . . . . CALL FOR PRICE
TELETYPE 43 KSR/WRS232 1087
CENTRONICS
730-1 parallel interface . . . . NEW LOW $ 599
730-3 RS232-C 649
737-1 parallel interface 799
737-3 RS232-C 849
739 w/graphics, List $999 829
779 w/tractor, List $1350 799
RS232 Serial 1595
704-9 w/tractor, parallel, VFU,
List $2350 1695
TI 810 Basic, List $1895 1595
810/serial & Centronics-style parallel interface, List $1940 1635
810 w/full ASCII (U/LC), VERTICAL Forms Control, Compressed Print 1795
TI 820 RO, Basic, lower case 1695
TI 820 KSR, List $2165 1795
TI 745, List $1695 1399
AXIOM IMP I 699
OKIDATA Microline 80, List $949 499
RS232 Serial Interface 99
tractor feed option 109
OKIDATA Microline 82 679
OKIDATA Microline 83 1069

NORTH STAR HORIZON
CALL

FLOPPY DISK SYSTEMS
NORTH STAR MDS-A-D, List $899 . . . . $ 749
MDS-A-Q, List $1099 898
THINKER TOYS Discus 2D
Discus 2D, single drive,DD,List$1095 . . $ 898*
Discus 2D, dual-drive,List$1875 . . . . $1549*
Discus 2&2, A&T,
DD, double sided, List $1395 . . . . $1169
Dual Discus 2&2, A&T, List $2495 . . . . . $2099*
*Now includes CP/M 2.2 and Microsoft Basic

HARD DISKS



M10, 10 Mbytes, List $3695 . . . . $3095*
M26, 26 Mbytes, List $4495. NEW LOW $3795*

PROM PROGRAMERS
SSM PB1 Kit . . . . $ 152
SSM PB1, A&T, List 225
SD Computer Prom 100 Kit 199
SD Computer Prom 100, A&T, List 264

FLOPPY DISK CONTROLLER BOARDS
Cromemco 16FDC, double density disk controller, List $595 . . . . $ 499
NORTH STAR, DD Controller, List $565 . . . 479
Disk Jockey 2D, A&T, List $399 329
SD Versafloppy I,Kit, List $284 239
SK Versafloppy I, A&T, List . . . . CALL
Versafloppy II, DD Kit, List $413 349
Versafloppy II, A&T, List . . . . CALL
DELTA double density, A&T 345
CONDUCTOR, double density, A&T 269
INTERSYSTEMS FDC-2, A&T . . . . CALL
MICROMATION Doubler, DD A&T 429
TARBELL Floppydisk, SD, Interface Kit . . . 199
double density, A&T, List $495 444

ESCON CONVERSION FOR IBM SELECTRIC
Complete with microprocessor controller and power supply. Factory built. User installs solenoid assembly or it can be done at Escon factory for $100.
Parallel (TRS-80, Sorcerer, etc.) $575 . . . $ 514
RS232 Serial, List $599 534
TRS-80 Cable 25

CALIFORNIA COMPUTER SYSTEMS
Z80 CPU BOARDS, List $299 . . . . $ 269
DISK CONTROLLER 2422,
with CPM, List $399 359
16K STATIC, A&T, List $349. 95 259
32K STATIC, List $710 599
64K DYNAMIC BOARD, List $699 499

CPU BOARDS
(assembled unless noted)
NORTH STAR Z80A
(ZPB-A/A), List $299 . . . . $ 254
CROMEMCO 4MHz (ZPU-W), List $395 . . . 335
4MHz (SCC-W), List $495 382
INTERSYSTEMS (formerly Ithaca Audio)
MPV-80, 4 MHz, List $395 349
SSM CBI 8080, A&T, List $252 214
CB2 Z-80, A&T, List $344 289
CB2 Kit, List $260 219
DELTA Z-80, with I/O 289
SD SBC-100 ASM, List $413 349
SBC-100 KIT, List $341 289
SBC-200 ASM, List $471 399
SBC-200 Kit, List $373 317

MEMORY BOARDS
32K SD EXPANDO RAM KIT
CALL FOR PRICES
NORTH STAR 16K Dynamic RAM Board
A&T (RAM)-16-A/A, List $499 . . . . $ 399
32K A&T (RAM-32/A), List $739 599
CROMEMCO 16KZ-W, List $495 419
64 KZ-W, List $1495 1269
MEMORY MERCHANT
16K Static-4 MHz 159
MEASUREMENT SYSTEMS & CONTROLS
(Guaranteed performance, inclu. labor/parts 1 yr.)
DM6400 64K Board w/all 64K, List $795 . . . 599
DM4800 with 48K, List $695 650
DM3200 with 32K, List $595 509
DMB6400 64K Baord w/all 64K 799
DMB4800 with 48K 789

INTERSYSTEMS
(formerly Ithaca Audio)
8K Static 2 MHz, A&T, List $165 . . . . $ 149
64K Dynamic, List $995 849

CALIFORNIA COMPUTER
16K Static, A&T, List $349.95 . . . . $ 269

VIDEO BOARDS
I/O Mapped
SD COMPUTER VDB-8024,
Kit, List $437 . . . . $ 369
Assembled, List $556 469
XITEX SCT-100K, Kit . . . . ONLY 169.95
SCT-100A, Assembled, List $215 . . . . . . 189.95
SSM VB2 I/O, Kit, List $199 169
Assembled & Tested, List $269 229

VIDEO BOARDS
(continued)
MEMORY MAPPED
SSM VB1C, 16x64, Kit, List $179 . . . . . . *$ 152*
Assembled & Tested, List $242 206
SSM VB3, 80-Char, 4 MHz,
Kit, List $425 . . . . 359*
4 MHz, A&T, List $499 424
*Subject to change

NEW CROMEMCO 16FDC DOUBLE DENSITY DISK CONTROLLER
List $595 . . . . OUR PRICE $499

F.O.B. shipping point. All prices subject to change and all offers subject to withdrawal without notice. Advertised prices are for prepaid orders. Credit card and C.O.D. 2% higher. C.O.D. may require deposit.

— WRITE FOR FREE CATALOG —

» Mini Micro Mart, Inc. «
943 W. Genesee St. Syracuse, New York 13204 (315) 422-4467

238

By Martin Klaver

Changing Times

Will Computers Soon Outnumber Telephones?

"The appearance of cheap, powerful and versatile microcomputers is a technical change with enormous and widespread implications, comparable in importance to the harnessing of electricity or even to the invention of steam engines." These are the words of Sir Charles Carter, chairman of the advisory committee of the Policy Studies Institute of London (from "Microprocessors in Manufactured Products," Policy Studies Institute, London, England).

To many, these words will doubtless be met with a good deal of resistance. The full implications of great technological changes are not usually apparent to all in the early stages of development.

Many managers view the microcomputer as just a new version of drab old data processing systems used to calculate payrolls. They do not yet apprehend the potential from placing problem-solving capabilities in the hands of individuals at very low cost. Since most jobs nowadays involve mental exercises, microcomputers will change most of our jobs, and possibly eliminate much of our work. This means that if you don't know how to use one, you may become obsolete at your work.

Even for those managers who do apprehend some of the potential of small microcomputers, it may not be altogether attractive to them to put all that problem-solving capability in the hands of some subordinate who may eclipse them in accomplishment.

Barriers

The most important obstacle to general acceptance is lack of knowledge about microcomputers. The new technology has developed so fast that chip is still a strange *word to many* otherwise well-informed people, and even a programmer working in a large data processing system may not know much about the new small models which have so suddenly come on the scene.

In contrast, those coming out of school, to whom the microcomputer may have actually been a teacher, may find a programming language part of their everyday language and regard the microcomputer as they do the telephone—an irreplaceable part of their lives.

Before purchasing a computer, you must invest a great deal of time in learning about your purchase, and many managers may not be willing or able to invest that time.

Probably the biggest threat to a manager is that microcomputers demand a new literacy he may not possess. Some will never master the language of computers.

Rather than face this fact, some managers may regard the microcomputer as just a game machine, an impression unfortunately shared by part of the microcomputer industry itself.

Of course, playing games with computers is a valuable learning experience, too. The effectiveness in cutting costs in a business and getting the most for your investment in computers depends precisely upon just how much you know and have used computers. It takes a while to become well informed and to understand the capabilities, and many managers will never make the effort needed to become competent enough to make intelligent choices.

Above all, anyone who really wants to understand what he can do with a microcomputer cannot avoid learning at least one computer language, especially BASIC or Pascal. Without a language you cannot know enough of computer capabilities to make good use of a computer. Dependence upon others for this expertise may cost heavily and lessen effectiveness.

To illustrate the importance of learning about microcomputers, ask yourself the following question: "In what other field has this absurd turnabout occurred: Two-to-five-year-old publications are selling for premium prices, while two-to-five-year-old equipment is going for a fraction of its original price?" Recently I could have bought a large, slightly used, heavy-duty printer for a central data processing system at less than the cost of two dozen copies of a leading publication covering two years. I could have bought a brand new printer for only $200. Of course, the old unit was obsolete and nowhere near as useful to me as a new 1981 printer. I would have needed a truck to get the older printer home.

The information on how to use the new microcomputer hardware has become more valuable than the equipment it is replacing. What the micros have done to large equipment value is a little like what small cars have done to large car values.

Any manager who has begun to read on the subject of microcomputers is on the right track. But, here again, there are obstacles to discourage the manager. Many books are badly written. Some computers have poor documentation. However, I have found that there is plenty of excellent material available, and if you talk to students and teachers and others browsing for books, the learning experience will be made easier and the pitfalls will be avoided.

It is well to note that one book on a computer language probably will not be enough. Since the applications of a computer language are so diverse, two books on BASIC may have only 25 percent in common.

A more immediate barrier to widespread use of microcomputers is the reluctance to try one out and see what it will do. Obviously, if a manager does not work with a microcomputer he cannot know what it will do for him.

Prestige

Another important factor in accepting microcomputers is prestige. Prestige can

(continued on page 238)

Martin Klaver (108 Eastmont Lane, Sicklerville, NJ 08081) is an industrial engineer responsible for special projects and planning.

Compare our prices with any in the magazine.

We win.

INTEGRATED COMPUTER SYSTEMS

ITHACA INTERSYSTEMS	CALL
ALTOS	CALL
ZENITH Z89	CALL
CALIF. COMPUTER SYSTEMS	CALL
MORROW DESIGNS	CALL

16 K RAM KITS

NEC 4116 200 ns	17.95

DISKETTES

MD 525-01 10 16	26.50
MD 550-01 10 16	44.50
MD 577-01 10.16	34.80
MD 557-01 10.16	45.60
FD 32 OR 34 -9000	36.00
FD 32 OR 34 -8000	45.60
FD 34-4001	48.60

VISTA COMPUTER CO.

APPLE 40 Track Drive A40 (163 K Bytes)	389.00
APPLE 80 Track Drive A80 (326 K Bytes)	549.00
APPLE 160 Track Drive A160 (652 K Bytes)	799.00
APPLE 80 Column Card	329.00
APPLE 8 Inch Disk Drive Controller	549.00

PRINTERS

ANADEX DP 9000	1195.00
ANADEX DP 9001	1195.00
ANADEX DP 9500	1295.00
ANADEX DP 9501	1295.00
CENTRONICS 739	765.00
C-ITOH 25 CPS PARALLEL	1440.00
C-ITOH 25 CPS SERIAL	1495.00
C-ITOH 45 CPS PARALLEL	1770.00
C-ITOH 40 CPS SERIAL	1870.00
C-ITOH TRACTOR OPTION	175.00
EPSON MX-80	$CALL
EPSON MX-80 F/T	$CALL
EPSON MX-100 F/T	$CALL
IDS-445G PAPER TIGER	795.00
IDS-460G PAPER TIGER	1149.00
IDS-560G PAPER TIGER	1495.00
INFOSCRIBE 500 9X9. 150 CPS (TI-810 REPLACEMENT)	1495.00
MALIBU 165 PARALLEL PRINTER	1995.00
MALIBU 165A PARALLEL PRINTER (Apple)	1995.00
MALIBU 200 DUAL MODE	2995.00
NEC SPINWRITER 5510 SERIAL RO	2595.00
NEC SPINWRITER 5530 PARALLEL RO	2595.00
NEC SPINWRITER 5500 D SELLUM OPTION	2795.00
NEC SPINWRITER 3500 SELLUM OPTION	2195.00
OKIDATA MICROLINE 80	399.00
OKIDATA MICROLINE 82	599.00
OKIDATA MICROLINE 83	795.00

MODEMS

NOVATION CAT ACOUSTIC MODEM	145.00
NOVATION D-CAT DIRECT CONNECT MODEM	155.00
NOVATION AUTO-CAT AUTO ANSWER MODEM	229.00
NOVATION APPLE-CAT	369.00
UDS 103LP DIRECT CONNECT MODEM	175.00
D.C. HAYES MICROMODEM II (Apple)	299.00
D.C. HAYES 100 MODEM (S-100)	319.00
D.C. HAYES STACK Smart Modem (RS 232)	249.00
LEXICON LX-11 MODEM	115.00

APPLE HARDWARE

VERSA WRITER DIGITIZER	219.00
ABT APPLE KEYPAD	119.00
MICROSOFT Z-80 SOFTCARD	299.00
MICROSOFT RAMCARD	170.00
ANDROMEDA 16K CARD	170.00
VIDEX 80 X 24 VIDEO CARD	299.00
VIDEX KEYBOARD ENHANCER	99.00
M & R SUPERTERM 80 X 24 VIDEO BOARD	305.00
NEC 12" GREEN MONITOR	235.00
SANYO 9" MONITOR (B & W)	179.00
SANYO 12" MONITOR (B & W)	230.00
SANYO 12" MONITOR (GREEN)	240.00
SANYO 13" COLOR MONITOR	429.00
TEECO 12" HIGH RES GREEN MONITOR	179.00
SSM AIO BOARD (INTERFACE) A&T	165.00
SSM AIO BOARD (INTERFACE) KIT	135.00
SSM IEEE 488 INTERFACE	369.00

MOUNTAIN HARDWARE

CPS MULTIFUNCTION BOARD	229.00
SUPERTALKER SD200	259.00
ROMPLUS WITH KEYBOARD FILTER	179.00
ROMPLUS W/O KEYBOARD FILTER	130.00
KEYBOARD FILTER ROM	49.00
COPYROM	49.00
MUSIC SYSTEM	459.00
ROMWRITER	149.00
APPLE CLOCK	239.00
A/D + D/A (DIGITAL TO ANALOG)	299.00
EXPANSION CHASSIS	575.00

CALIF. COMPUTER SYSTEMS

S-100 BOARDS

2200A MAINFRAME	349.00
2032A 32K STATIC RAM	599.00
2065C 64K DYNAMIC RAM	499.00
2422 FLOPPY DISK CONTROLLER & CP/M	339.00
2710 FOUR SERIAL I/O	245.00
2718 2 SERIAL/2 PARALLEL I/O	265.00
2720 FOUR PARALLEL I/O	189.00
2810 Z-80 CPU	249.00

APPLE BOARDS

7710A/D ASYNCHRONOUS SERIAL INTERFACE	139.00
7712A SYNCHRONOUS SERIAL INTERFACE	149.00
7424A CALENDAR/CLOCK	99.00
7728A CENTRONICS PRINTER INTERFACE	99.00
7720A PARALLEL INTERFACE STD FIRMWARE	99.00

APPLE SOFTWARE

MAGIC WINDOW WORDPROCESSOR STD APPLE	89.00
MAGIC WAND (REQUIRES Z80 SOFTCARD & 80 COL)	275.00
WORDSTAR—APPLE (REQUIRES Z80 SC & 80 COL)	275.00
MAILMERGE—APPLE (REQUIRES WORD STAR)	90.00
MICROSOFT FORTRAN (REQUIRES Z80 SOFTCARD)	165.00
MICROSOFT COBOL (REQUIRES Z80 SOFTCARD)	550.00
DB MASTER 2.4	159.00
VISICALC 16 SECTOR	189.00
CCA DATA BASE MANAGER	99.00
A-STAT COMPREHENSIVE STATISTICS PKG	119.00

CP/M SOFTWARE

MICROSOFT BASIC-80	299.00
MICROSOFT BASIC COMPILER	319.00
MICROSOFT FORTRAN-80	399.00
PEACHTREE SYSTEMS	CALL
MAGIC WAND (REQUIRES CP/M)	275.00
WORD STAR (REQUIRES CP/M)	310.00
MAILMERGE (REQUIRES WORD STAR)	100.00
SPELLGUARD	239.00
CP/M PICKLES & TROUT FOR TRS-80 MOD II	175.00

Alpha Byte STORES

We built a reputation on our prices and your satisfaction.

We guarantee everything we sell for 30 days. If anything is wrong, just return the item and we'll make it right. And, of course, we'll pay the shipping charges.

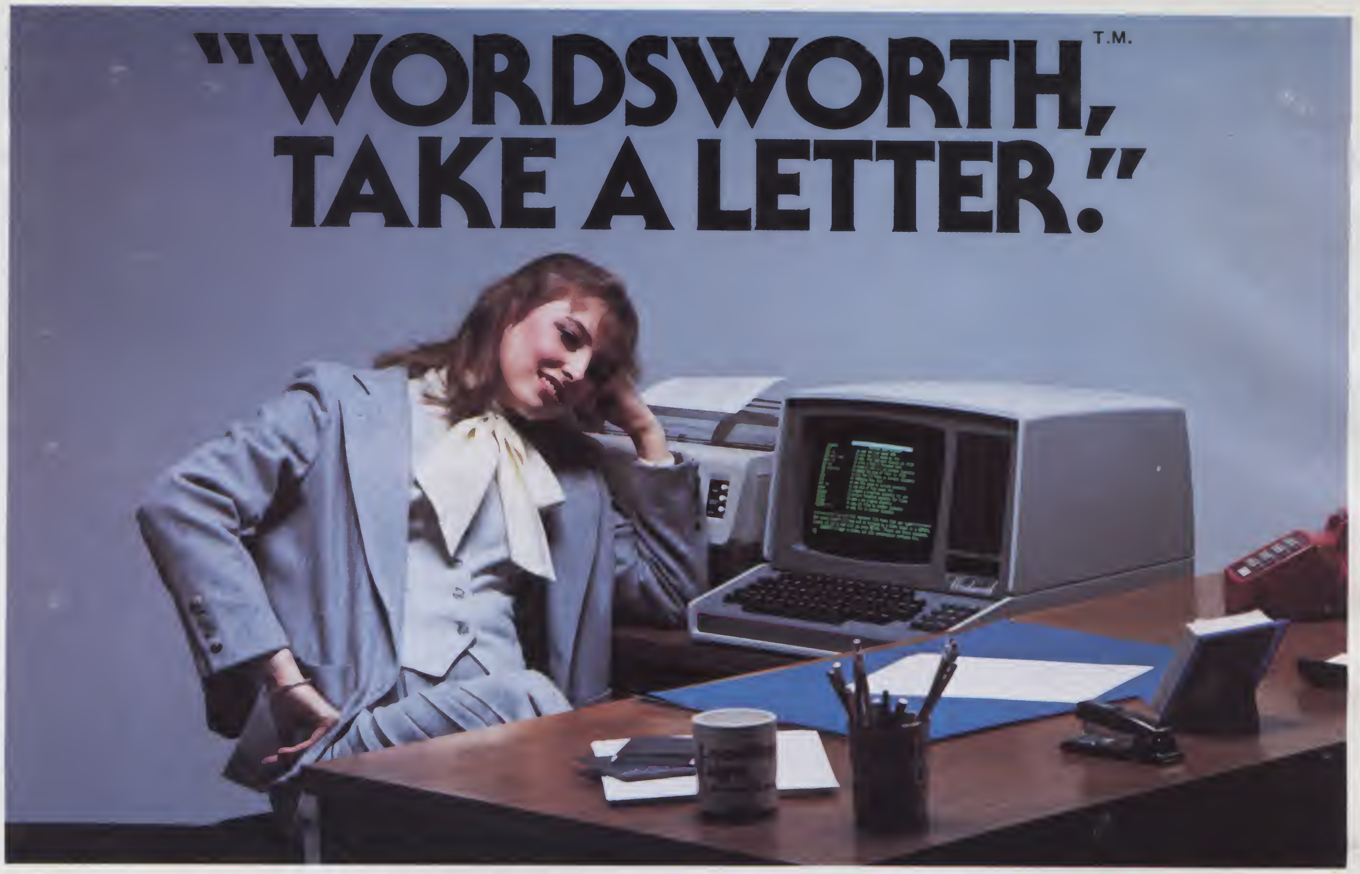
We accept Visa and Master Card on all orders. COD accepted up to $300.00. We also accept school purchase orders.

Please add $2.00 for standard UPS shipping and handling on orders under 50 pounds, delivered in the continental U.S. Call us for shipping charges on items that weigh more than 50 pounds. Foreign, FPO and APO orders please add 15% for shipping. California residents add 6% sales tax.

(213) 883-8594

31245 La Baya Drive, Westlake Village, California 91362 311

The prices quoted are only valid for stock on hand and all prices are subject to change without notice.



THE FIRST SUPER-SIMPLE LETTER QUALITY WORD PROCESSOR IN THE WORLD THAT CAN SIT NEXT TO YOUR SECRETARY— FOR LESS THAN $5,000.

"Wordsworth™" removes the fear and loathing many people have about things called "computers" and "word processing." Because Wordsworth just sits at a desk and does what he's told. With unique new, simplified software, it can tell its typist what to do—step by simple step—to perform even the most complicated tasks.

It can not only "take a letter," but it can revise it, customize it in millions of ways (quite literally), personalize it, print it on your letterhead or business form so you can't tell it from hand-typed . . . in short, everything but put it in an envelope.

Moreover, it can perform all sorts of mundane business functions, like general ledgers and accounting reports, using easily available industry-compatible CP/M™ software.

Installation? Plug it in. (Cable supplied.)

Size? Bigger than a breadbox—but not by much. It's about half as big, or less, than systems that can't do half as much. Definitely desktop—a sub-compact 20″ by 40″.

Service? Available everywhere.

Price? About $4,995. Complete and ready to go.

For the name of your nearest dealer—and a free hands-on demonstration—just pick up the nearest telephone.

TOLL-FREE 1-800-343-6833.

In Massachusetts, call collect (617) 828-8150. Telex 951-624.

LEADING EDGE.™ 355

Leading Edge Products, Inc., 225 Turnpike Street, Canton, Massachusetts 02021
DEALERS: For immediate delivery from the Leading Edge Inventory Bank,™ on "Wordsworth" and other of the industry's most popular products, just give us a call.